スタイルシート辞典 第3版

Interner Explorer6.0 & Netscape6.2

(株)アンク著

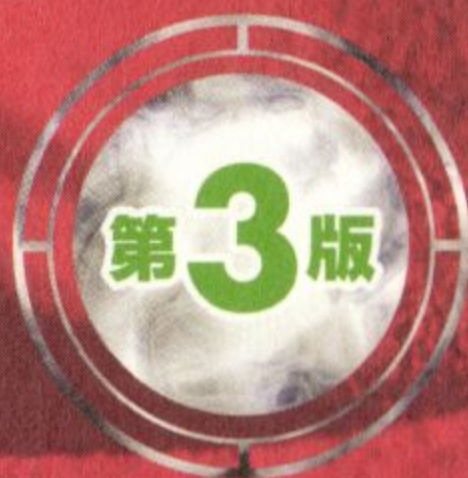
SE
SHOEISHA

本書は、好評を博した
「スタイルシート辞典 第2版」の改訂版です。

最新ブラウザInternet Explorer 6.0＆
Netscape 6.2に対応し、
豊富な画面と見やすい紙面で、
スタイルシート（CSS）の基本から
やさしく解説しています。
巻末には配色＆ビジュアルサンプルもついており、
デザインのヒントとしても利用できます。
まるごと１冊CSSは本書だけ！
CSSの純正解説書として、
ホームページ初心者からWebデザイナーの方まで、
長く手元に置いていただける１冊です。

本書のサンプルソースはすべて
Webページからダウンロードが可能です
さあ！ 今すぐ自分のページに活かしましょう!!

http://www.shoeisha.com/book/pc/dic/

スタイルシート辞典 第3版

Interner Explorer6.0 & Netscape6.2

(株)アンク著

本書内容に関するお問合せについて

このたびは翔泳社の書籍をお買い上げいただき、誠にありがとうございます。弊社では、読者の皆様からのお問い合わせに適切に対応させていただくため、以下のガイドラインへのご協力をお願い致しております。下記項目をお読みいただき、手順にしたがってお問い合わせください。

●ご質問される前に

弊社Webサイトの「Q&Aコーナー」(http://www.shoeisha.com/info/help.asp)をご参照ください。これまで受けたご質問への回答(FAQ)や、的確なご質問方法に関する情報を掲示しています。

●ご質問方法

弊社Webサイトの書籍専用質問フォーム(http://www.shoeisha.com/book/qa/)をご利用ください。記載漏れや独自の用紙等によるご質問、お電話や電子メールによるお問合せ、本書にはさみ込まれたアンケートはがきに記入されたご質問等は、お受けしておりません。

※質問専用シートのお取り寄せについて

Webサイトにアクセスする手段をお持ちでない方は、ご氏名、ご送付先(ご住所/郵便番号/電話番号またはFAX番号/電子メールアドレス)および「質問専用シート送付希望」と明記のうえ、電子メール(qaform@shoeisha.com)、FAX、郵便(80円切手をご同封願います)のいずれかにて“編集部読者サポート係”までお申し込みください。お申し込まれた手段によって、折り返し質問シートをお送りいたします。シートに必要事項を漏れなく記入し、“編集部読者サポート係”までFAXまたは郵便にてご返送ください。

●ご回答について

ご回答は、ご質問いただいた手段によってご返事申し上げます。ご質問の内容によっては、回答に数日ないしはそれ以上の期間を要する場合があります。

●ご質問に際してのご注意

本書の対象を越えるもの、記述個所を特定されないもの、また読者固有の環境に起因するご質問等にはお答えできませんので、予めご了承ください。

●郵便物送付先およびFAX番号

送付先住所　〒160-0006　東京都新宿区舟町5

FAX番号　03-5362-3818

宛先　(株)翔泳社出版局 編集部読者サポート係

※本書に記載されたURL等は予告なく変更される場合があります。

※本書の動作環境に関する詳細はviiページをご参照ください。

※本書の出版にあたっては正確な記述につとめましたが、著者や出版社などのいずれも、本書の内容に対してなんらかの保証をするものではなく、内容やサンプルに基づくいかなる運用結果に関してもいっさいの責任を負いません。

※本書に掲載されているサンプルプログラムやスクリプト、および実行結果を記した画面イメージなどは、特定の設定に基づいた環境にて再現される一例です。

CONTENTS

CONTENTS

第1部　スタイルシートの基礎

第2部　スタイルシートリファレンス

テキスト

付　録

INTRODUCTION

本書の動作環境

本書は以下の環境におけるブラウザ表示に基づいて記述されています。

日本語版 Microsoft Windows XP
　Windows 版 Internet Explorer 6.0
　Windows 版 Netscape 6.2

サンプルソースを表示しているディスプレイ画面は、基本的に各ブラウザのデフォルト設定（初期設定）ですが、効果が明確にあらわれるように

Internet Explorer 6.0　［文字サイズ］＝［最大］
Netscape 6.2　フォント［サイズ］＝［20］ピクセル

に設定しています。

なお、サイズを変えたほうが効果が明確にあらわれると判断した項目は、適宜設定を変更しています。

フォントはデフォルトのままですので Internet Explorer 6.0、Netscape 6.2 ともに「MS P ゴシック」となります。

この設定はあくまでも一例ですので、ユーザーのフォントサイズやフォントの種類によって必ずしも本書の表示通りにはならないことをご了承ください。

INTRODUCTION

本書の読み方

第2部「スタイルシートリファレンス」では、スタイルシートの効果や利用する場面にあわせて9つのカテゴリに分けて解説しています。

各項目のタイトルは「文字色を指定したい」など、スタイルシートの機能を目的から引ける形式になっています。

各項目の構成要素は基本書式・解説・サンプルソース・サンプルソースを表示した画面となっており、項目によってはコラムやHTMLタグに書き換えた場合のメモも掲載しています。

●カテゴリ
効果・場面によって分かれています

●タイトル
具体的に何ができるかを表しています。スタイルシートの使用目的から選んでください

●基本書式
その項目で解説するプロパティの基本書式です。解説するプロパティを赤色、値を青色で表記しています。なお、本書ではプロパティ・値ともにすべて小文字で表記しています

●値
そのプロパティがとる値で、基本書式では★や☆で表しています。値の詳細については本文中で解説しています

●解説
プロパティや値についての解説です

●対応ブラウザアイコン
その項目で解説しているスタイルシートが対応しているブラウザ（Windows版Internet Explorer 6.0、Netscape 6.2）をアイコンで示しています。アイコン表記のない場合は、そのブラウザが対応していないことを示します

4 FONT

フォントを斜体にしたい

フォント

font-style: ★

★……キーワード

文字を斜体にします。
値には以下のキーワードがあります。

italic	斜体
oblique	斜体
normal	通常の状態で表示（デフォルト）

厳密な定義によればitalicとobliqueは異なります。しかし、現在のところ一般的なブラウザではその違いを区別しておらず、同じように表示されます。

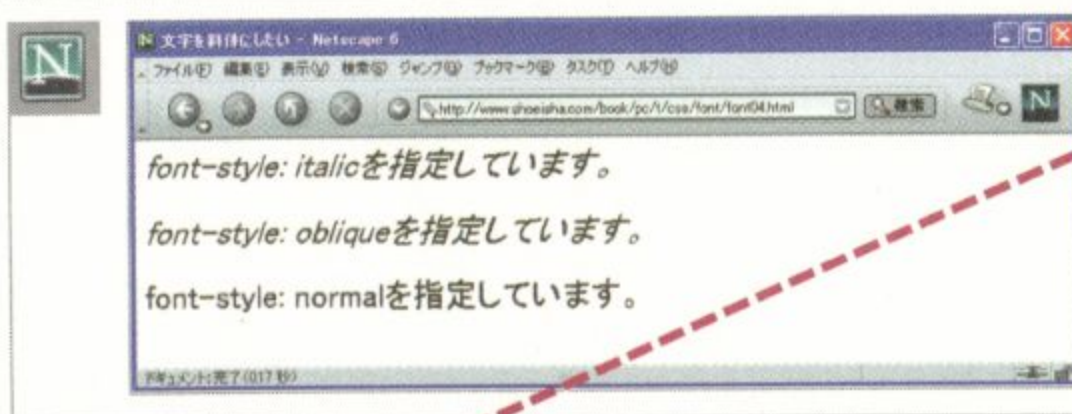
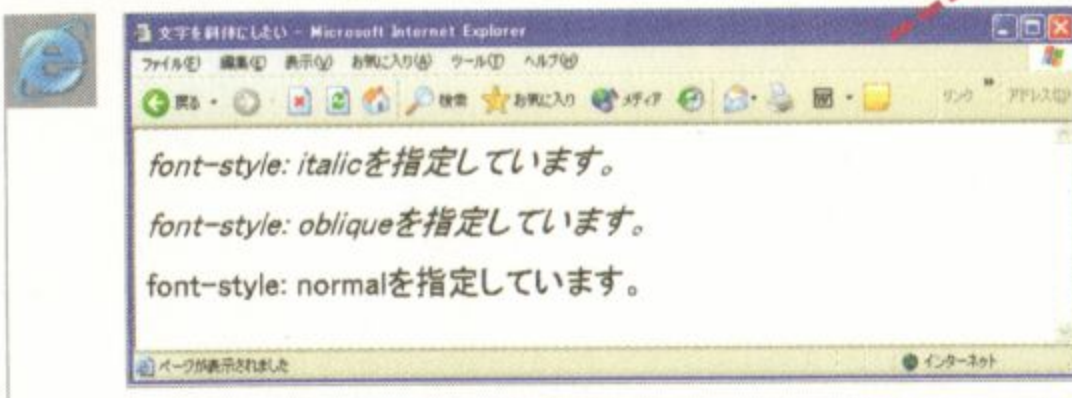
SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title> 文字を斜体にしたい </title>
<style type="text/css">
<!--
body        { font-size: 20pt }
p#sample1   { font-style: italic }
p#sample2   { font-style: oblique }
p#sample3   { font-style: normal }
-->
</style>
</head>
<body>
```

126 フォントを斜体にしたい

●サンプルソース

その項目で解説しているプロパティや値を使用したサンプルソースです。
解説しているプロパティは赤色、値は青色、該当するセレクタやそれを受けるタグについては緑色で表記しています。本書では、すべてHTML文書内でスタイルシートを設定する方法をとっています（p.15参照）。また、classとidの使い分け（p.25参照）は、サンプルに多様性を持たせるため、特に規則性はありません。id指定とclass指定を置き換えても基本的には同様の結果となります。
なお、紙面の関係で一部省略したり、改行を行っています

```
<p id="sample1">font-style: italicを指定しています。</p>
<p id="sample2">font-style: obliqueを指定しています。</p>
<p id="sample3">font-style: normalを指定しています。</p>
</body>
</html>
```


斜体を指定するHTMLタグをCSSに改める

HTMLタグで斜体を指定するには、<i>タグを利用します。
<i>～</i>

同様の効果をスタイルシートで表現すると以下のようになります。
★ { font-style: ☆ }　（★――セレクタ　☆――italicまたはoblique）

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
italic	○	○	○	○	○	○	○
oblique	○	○	○	○	×	×	○
normal	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照
文字を装飾したい……p.60
フォントを一括して指定したい……p.130

●ディスプレイ

サンプルソースを実際にブラウザで表示した場合の画面です。対応していないブラウザにはアイコンに×をつけています

●コラム

その項目のスタイルシートを使用する際の注意点や関連するトピック、さらに理解を深めるための内容を掲載しています

●～HTMLタグをCSSに改める

解説している項目のスタイルシートをHTMLタグで記述した場合を参考として紹介しています。
デザインに関する指定はスタイルシートの使用が推奨されているので、書き換えの際の参考になります

●対応表

旧バージョンにおける各ブラウザの対応表です（Windows版）。Macintosh版での動作が異なる場合など、特筆すべき点は欄外に明記しています

●参照

関係の深い項目へのリンクです。合わせて参照することで体系的に理解が深まります

第1部

スタイルシートの基礎

STYLESHEET BASIC

1 STYLESHEET BASIC

スタイルシートとは

Webページの話題のなかで、スタイルシート・CSSという言葉は決してめずらしいものではなくなりました。スタイルシートとは、ひとことで表現するならば「Webページのレイアウトを定義する技術」ということができるでしょう。

Webページを記述するHTMLは、文書の論理的な構造を示す言語です。たとえば、見出しがあり、本文の中に段落やリストがあり……といったように、文書がどのような要素で構成されているのか、またある特定の部分が文書全体の中でどのような意味をもっているのかを、コンピュータに知らせるための言語なのです。表現方法を指定するための言語ではありません。しかし実際は、Webの発展にともない、色やフォントサイズの指定、レイアウトのためのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するようになっていきました。W3Cではこの状況を改めるため、構造に関する指定と体裁に関わる指定とを分離させ、HTMLの機能ではない体裁の制御については別の方法を導入しようと考えるようになりました。こうした姿勢のもとに生み出されたのがスタイルシートの概念です。

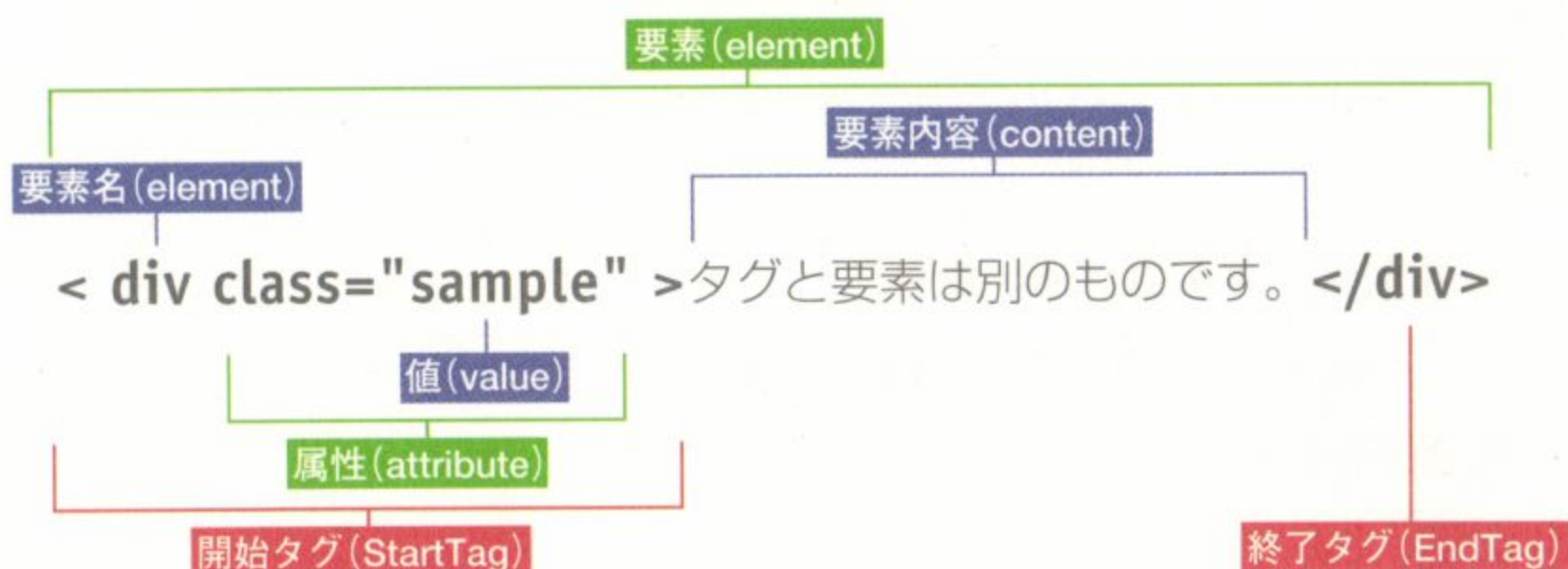
W3Cは1996年12月にスタイルシート言語の1つであるCSS1（Cascading Style Sheets Level 1）を勧告し、Internet Explorer 3.0とNetscape Navigator 4.0がこの技術を導入しはじめました。CSS1ではHTMLで可能だったデザインのほとんどを扱えるようになっています。その後1998年5月12日には次の規格であるCSS2が出されました。CSS2は、CSS1よりも細かい設定ができるようになっています。また、デザイン面だけでなく印刷や音声に関する設定など、Webページ以外への出力についても考慮されるようになりました。その背景にはXMLの登場があります。XMLはHTMLに比べてブラウザへの依存度が低く、HTMLでは実現できないさまざまな設定や操作が可能です。XMLでも、ページのデザインについてスタイルシートを利用します。本書はスタイルシートをHTMLに組み込むことに特化した内容になっていますが、基本を理解できればXMLへ実装することもそれほど難しいものではありません。ぜひ本書でスタイルシートをマスターしてください。

なお、現在一般的にいわれている「スタイルシート」とは、スタイルシート言語の1つであるCSSを指していることが多く、本書でもCSSをスタイルシートと表記しています。

2 STYLESHEET BASIC
HTML の基本

スタイルシートを組み込む上で必要となるHTMLの知識を確認しておきましょう。HTML文書の作成方法など、もっとも基本的な点については本書では割愛しますので、詳しくは本書姉妹書『HTMLタグ辞典第5版』やその他のホームページ作成書を参考にしてください。

● HTML の基本構造

HTMLの一番基本的な構造を示すと次のようになります。

このようにHTMLタグが形成するかたまりを「要素」といいます。要素は、より詳しくはブロックレベル要素とインラインレベル要素に大別され、他のどの要素を内容に含めることができるかなどに決まりがあります。たとえばインラインレベル要素のなかにブロックレベル要素を入れることはできません。

また、スタイルシートとの関連でみるならば文字の水平位置を指定するtext-alignプロパティはブロックレベル要素のみに適用、文字の垂直位置を指定するvertical-alignプロパティはインラインレベル要素とセル要素のみに適用することになっています。

定義から外れる要素であっても指定が有効な場合もありますので、本書は要素のタイプをあまり意識しなくても利用できますが、より正確にスタイルシートを記述するならば、要素がどんなタイプに属するかはぜひ理解しておきたい点です。

※本書では便宜上タグ、要素、要素名という言葉を併用し、状況に応じて使い分けながら解説をすすめます。これらの言葉は同じものと考えられがちですが、厳密には違うものですので注意して覚えてください。

ブロックレベル要素

見出しや段落など１つのまとまりを構成する要素です。一般的には前後に改行が入ります。

address, blockquote, center, dir, div, dl, fieldset, form, h1 〜 h6, hr, isindex, menu, noframes, ol, p, pre, table, ul

インラインレベル要素

文字と同じレベルで扱われる要素です。一般的には前後に改行は入りません。

a, abbr, acronym, applet, b, basefont, bdo, big, br, button, cite, code, dfn, em, font, i, iframe, img, input, kbd, label, map, object, q, s, samp, select, small, span, strike, strong, sub,sup, textarea, tt, u, var

※ ins 要素、del 要素は、ブロックレベル、インラインレベルの両方で使えます。

別扱いの要素

スタイルシートではブロックレベル要素のなかでも以下のものについては、別に扱う場合もあります。

テーブル要素、セル要素、キャプション要素

テーブル要素は <table> タグ、セル要素は <th> タグや <td> タグ、キャプション要素は <caption> タグの要素です。

リストアイテム要素

リストアイテム要素はリストの <li> タグ（リスト項目の１つ１つ）のことです。

大文字か小文字か

HTML 文書中で使われるタグや要素名、属性、スタイルシートの設定などは大文字小文字を区別しません。本書では XHTML がすべてを小文字で書くよう定義されていることを考慮して、小文字で表記するよう統一しています。

● HTML の木構造と親要素・子要素

HTML は文章の論理構造を示すものですから、HTML 文書は木構造（ツリー構造）のかたちをとっています。

どのような木構造となるかは、文書によって異なりますが、通常 <html> タグの下位には <head> タグと <body> タグがあり、<body> タグの下には文書の内容に応じて木構造が展開されることになります。この木構造において、１つ上位の階層にある要素（外側にある要素）を「親要素」といいます。逆に親要素から見てその階層下にある要素（内側に含まれる要素）は「子要素」となります。親要素から子要素、さらに子要素（これらをまとめて「子孫」と表現する場合もあります）というように木構造が展開されてゆくのです。木構造の出発点である html 要素はル

ート要素と呼びます。

たとえば下記のサンプルソースにおいて、 タグは <ul> タグのなかに含まれています。つまりこのHTML文書ではli要素の親要素はul要素となります。同じようにul要素の親要素はbody要素となり、逆にbody要素は子要素h1、ul、h2、pを持つということになります。

スタイルシートにおいては、このような親要素・子要素の関係によってセレクタの設定やデフォルト値、継承などに影響があります。親要素・子要素の関係を理解しておきましょう。

```
<html>
<head>
  <title>HTMLの木構造と親要素・子要素</title>
</head>
<body>
  <h1>HTMLの木構造と親要素・子要素</h1>
  <ul>
    <li>HTMLの木構造</li>
    <li>親要素・子要素</li>
  </ul>
  <h2>木構造とは</h2>
   <p>HTMLは文章の論理構造を示すものですから、HTML文書は<strong>木構造（ツリー構造）
   </strong>のかたちをとっています。</p>
</body>
</html>
```

▲サンプルソースの木構造

3 STYLESHEET BASIC

スタイルシートの書式

セレクタ { プロパティ: 値 }

スタイルシートの基本的な書式は次のようになります。

セレクタ	スタイルを適用する対象
宣言	設定するスタイル
プロパティ	設定するスタイルの性質（色、大きさなど）
値	プロパティごとに決められている値

このように、スタイルシートは「"セレクタ"の"プロパティ"を"値"にする」という形で設定し、HTML文書に組み込んでいくものです。

この例ではh1要素に対して色をブルーにするよう指定しています。このスタイルを設定した文書ではh1要素が出てきた場合、その範囲はブルーで表現されることになります。

つまり、セレクタとはスタイルを適用させる対象です。プロパティと値には、セレクタに対してどのような指定をするか、指定するスタイルの種類とその具体的な値を記述します。プロパティを複数設定する場合には{ }の中に、「;」（セミコロン）で区切って並べます。複数行になってもかまいません。その際、宣言の最後にセミコロンをつけることもできます。

では、上の例を利用して実際に簡単なソースを書いてみましょう。

```
<html>
<head>
<title>スタイルシートのテスト</title>
<style type="text/css">
h1      {
        color: blue;
        font-style: italic
}
```

```
</style>
</head>
<body>
<h1>スタイルシート</h1>
<h2>スタイルシートとは</h2>
<p>
Webページを記述するHTMLは、文書の論理的な構造を……
</p>
</body>
</html>
```

この例では、<style>タグを利用して、h1{ color: blue; font-style: italic }というスタイルを設定しています。その結果、文書中の<h1>タグで囲まれた範囲が、設定したようにブルー(color: blue)のイタリック体(font-style: italic)で表示されています。

<style>タグを利用するほかにも、スタイルシートの設定にはいくつか方法がありますが、詳しくは「スタイルシートの設定方法」(p.11)を参照してください。

4 STYLESHEET BASIC

コメントとコメントアウト

スタイルシートでコメントを入れるには、「/*」と「*/」で挟みます。コメントは入れ子にすることはできません。

また、前項の例のように<style>タグを使ってスタイルシートを設定すると、スタイルシートに対応していないブラウザでは、ソースがそのまま表示されてしまう可能性があります。この対策として、「<!--」と「-->」でスタイルの設定個所全体をコメントアウトして、その部分がそのまま表示されるのを防ぐ方法があります。

```
<html>
<head>
<title>スタイルシートのテスト</title>
<style type="text/css">
<!--
h1	{
	color: blue;			/* h1要素をブルーのイタリック体に設定 */
	font-style: italic
}
-->
</style>
</head>
<body>
<h1>スタイルシート</h1>
<h2>スタイルシートとは</h2>
<p>
Webページを記述するHTMLは、文書の論理的な構造を……
</p>
</body>
</html>
```

▲「/*」「*/」で挟まれた部分は表示されないので、メモ書き等に利用することができます

ソースを見やすくするには

プロパティ、コロン、値の前後には半角スペース、タブ、改行を入れることができるので、適宜入れて記述するとソースが見やすくなります。

スタイルシートにおける引用符

HTMLでは値を「" "」(ダブルクォーテーション)や「' '」(シングルクォーテーション)で囲むことが推奨されています。しかし、スタイルシートでは値として記述するキーワードをこうした引用符で囲うと、文字列として解釈されてキーワードではなくなり、スタイルシートの効果が得られなくなってしまいますので注意してください。

例:	red	キーワードとして認識される
	"red"	文字列として認識されてしまう

5 STYLESHEET BASIC

デフォルトのスタイルシートの設定

<meta http-equiv="Content-Style-Type" content="text/css">

現在一般的にいわれている「スタイルシート」とは、スタイルシートの一種であるCSS（Cascading Style Sheets）を指していることが多いのですが、実際にはスタイルシートの実現方法はCSSだけではありません。そのため、スタイルシートを利用する場合にはデフォルトで使用するスタイルシートを指定しておく必要があります。これには<meta>タグを使って指定します。必ず<head>タグと</head>タグの間に記述してください。

多くの場合はこれらの情報を記述しなくてもブラウザ側が自動的に判別しますが、文字化けや誤動作が生じないとも限りません。正しく表示させるためには指定しておくべきです。

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>スタイルシートの設定</title>
<style type="text/css">
<!--
        :
-->
</style>
</head>
<body>
        :
</body>
</html>
```

6 STYLESHEET BASIC

スタイルシートの設定方法

スタイルシートの設定には大きく分けて以下の３つの方法があり、スタイルシートを利用する状況に応じて使い分けができるようになっています。

1　外部スタイルシートを読み込む（p.12 参照）
2　HTML 文書にまとめて設定する（p.15 参照）
3　タグに直接スタイルを設定する（p.17 参照）

それぞれ適用される範囲が異なるだけでなく、競合した場合の優先順位もほぼ決まっている（p.19 参照）ことから、これらを組み合わせて柔軟にスタイルを指定することができます。

1　外部スタイルシートを読み込む

2　HTML 文書内にまとめて設定する

3　タグに直接指定する

● 1..外部スタイルシートを読み込む—— link 要素を使用する

<link rel="stylesheet" href="★" type="text/css">

★••••••スタイルファイルの URL（拡張子は *.css）

スタイルを設定したファイルをHTML文書とは別に用意し、これを<link>タグで読み込む設定方法です。<head>～</head>内に記述し、href属性で外部ファイルのURLを指定してください。この場合用意する外部ファイルは、スタイルのみを記述し、「*.css」という拡張子を付けたテキストファイルです。

rel属性ではスタイルシートにリンクしていることを、type属性ではそのリンクされているスタイルシートの種類（タイプ）が「text/css」であることを示します。スタイルシートの実現方法は（本書で解説している）CSSだけではありません。そのため、type属性で利用するタイプを指定しておく必要があるのです。

この方法はひとつのスタイルファイルをサイト全体に適用させ、サイト全体のページの雰囲気を統一したい場合などに利用するとよいでしょう。

なお、複数の外部スタイルシートを利用したい場合は、必要な数だけ<link>タグによる指定を記述します。

▲ひとつのスタイルファイルを複数のHTML文書に適用することができます

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>外部スタイルシートを読み込む</title>
<link rel="stylesheet" href="design.css" type="text/css">
</head>

<body>
<h1>スタイルシートとは</h1>
<p>
スタイルシートは、ひとことで表現するならば「Webページのレイアウトを定義する技術」
ということができます。文書の論理構造に関する指定と体裁に関わる指定とを分離させ、本
来HTMLの機能ではない体裁の制御については別の方法を導入しようという姿勢のもとに生
み出されました。
</p>
</body>
</html>
```

スタイルファイル（ファイル名はdesign.css）

```
h1      {
        background-color: #cfc;
        color: rgb(0,128,0)
}
p       { color: blue }
```

▲design.cssの指定が反映されます

▲design.cssを読み込む設定をしない場合の表示（スタイルシートは適用されません）

外部スタイルシートの利点

外部スタイルを使ってスタイルシートを設定すると、ひとつのスタイルファイルを複数のHTML文書で共有できるだけでなく、HTML文書のソースを修正せずにスタイルを変更することができます。また、スタイルファイルをいくつも用意して、必要に応じたスタイルファイルを読み込むように設定することもできます。このようにメンテナンスがしやすい、柔軟な表現ができる、といったスタイルシートの利点を活かすために、W3Cでは外部スタイルの使用が推奨されています。

● 2..HTML文書内にまとめて設定する――style要素を使用する

<style type="text/css"> ★ </style>

★……スタイルの設定

HTML文書に設定したいスタイルを記述する方法です。<style>と</style>タグの間でスタイルを定義し、これを<head>～</head>内に配置します。記述したページ内でのみ有効になるので、ページごとにスタイルを適用したい場合などに便利な方法です。

本書のサンプルソースは基本的にこの方法を使った例となっています。

▲HTML文書内のみで有効です。ほかのHTML文書には影響しません

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>HTML文書内にまとめて設定する</title>
<style type="text/css">
<!--
h1      {
        background-color: #cfc;
        color: rgb(0,128,0)
}
p       { color: blue }
-->
</style>
```

```
</head>

<body>
<h1> スタイルシートとは </h1>
<p>
スタイルシートは、ひとことで表現するならば「Webページのレイアウトを定義する技術」
ということができます。文書の論理構造に関する指定と体裁に関わる指定とを分離させ、本
来HTMLの機能ではない体裁の制御については別の方法を導入しようという姿勢のもとに生
み出されました。
</p>
</body>
</html>
```

● 3..タグに直接スタイルを設定する——style属性を使用する

<☆ style="★"> ~ </☆>

☆••••••スタイルを設定したいタグ
★••••••スタイルの設定

style属性を使用し、タグに直接スタイルを記述する方法です。この場合はstyle属性の値としてスタイルを設定するため、「style=」につづけて「" "」の中にプロパティと値を記述していきます。特定の箇所にのみスタイルを設定する場合に利用します。

▲style属性で指定したタグにのみ有効になります

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>タグに直接スタイルを設定する</title>
</head>

<body>
<h1 style="background-color: #cfc; color: rgb(0,128,0)">スタイルシートとは
</h1>
<p style="color: blue">
スタイルシートは、ひとことで表現するならば<span style="color: red">「Webページ
のレイアウトを定義する技術」</span>ということができます。文書の論理構造に関する
指定と体裁に関わる指定とを分離させ、本来HTMLの機能ではない体裁の制御については別
の方法を導入しようという姿勢のもとに生み出されました。
```

```
</p>
</body>
</html>
```

▲タグの属性としてスタイルシートを指定すると、タグごとに細かい設定が可能になります

7 STYLESHEET BASIC

スタイルの優先順位

スタイルシートでは、外部スタイルシートを読み込む方法、style要素を使用してHTML文書内にまとめて設定する方法、style属性を使用してタグに直接スタイルを設定する方法などの設定方法が用意されています。そのため、1つの文書に複数のスタイルが設定され、それらが競合する可能性もあります。こうした場合に対処するためにスタイルには基本的な優先度がつけられています。

style属性による設定　　style要素による設定　　外部スタイルによる設定

高い　　優先順位　　低い

つまり、後からブラウザに読み込まれるスタイルほど優先順位が高くなる（後から読み込まれた設定が先に読み込まれた設定を上書きする）というわけです。また、1つの文書内で競合がおこった場合は、より限定的で詳細な指定を行っているスタイルが優先されます。たとえば要素に対して指定したスタイルよりは、クラスによるスタイルのほうが優先される、といったようになります。

スタイルシートの制作者による優先順位

スタイルシートは文書を制作した人によるものだけでなく、ブラウザが初期設定として持っているスタイルシートやユーザーが定義するスタイルシートなどもあります。このように制作者の異なるスタイルが同時に設定された場合、それぞれの優先度は次のようになります。

文書制作者のスタイル　　ユーザーのスタイル　　ブラウザのスタイル

高い　　優先順位　　低い

実際にはこうした条件も加わって最終的な優先順位が処理されるのです。

さらに、「!important」というキーワードを指定しておくことで、これらの優先度を逆転させることもできます（p.20参照）。

最優先のスタイルを設定する

★ { ▲: △ !important }

★••••••セレクタ
▲••••••プロパティ
△••••••値

スタイルシートには制作者や設定方法によって優先順位がつけられて処理されます（p.19参照）。しかし同時に複数のスタイルシートを適用することで、場合によってはスタイルが競合し、意図したスタイルにならない可能性も生じます。こうした問題を避けるため、優先させたいスタイルに「!important」というキーワードを指定しておく方法があります。

「!important」は優先させたいスタイルの「プロパティ:値」に続いて指定します。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>最優先のスタイルを設定する</title>
<style type="text/css">
<!--
h1      {
        background-color: #cfc;
        color: rgb(0,128,0) !important
}
p       { color: blue !important }
-->
</style>
</head>

<body>
<h1 style="color: #000">スタイルシートとは</h1>
```

```
<p style="color: orange">
スタイルシートは、ひとことで表現するならば「Webページのレイアウトを定義する技術」
ということができます。文書の論理構造に関する指定と体裁に関わる指定とを分離させ、本
来HTMLの機能ではない体裁の制御については別の方法を導入しようという姿勢のもとに生
み出されました。
</p>
</body>
</html>
```

▲!importantを指定していると、h1要素ではcolor:#000よりもcolor:rgb(0,128,0)が、p要素ではcolor:orangeよりもcolor:blueが優先されます

▲!importantを指定しないと、スタイルの優先順位によってcolor:#000とcolor:orangeが優先されます

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
	○	○	○	○	×	×	○

8 STYLESHEET BASIC

セレクタの種類

「スタイルシートの書式」（p.6参照）ですでに述べたように、セレクタとはスタイルを適用する対象を指定するものです。「スタイルシートの書式」ではタグにスタイルを設定するために、要素名をセレクタとして利用しました。実際にはセレクタの指定方法にはいくつもの種類があり、スタイルシートを利用する状況に応じて使い分けができるようになっています。主に利用されるのは、次のようなセレクタです。

1. タグにスタイルを設定する（p.23参照）
2. 任意の範囲にスタイルを設定する（p.25参照）
3. タグの相関関係（子孫の要素）でスタイルを設定する（p.28参照）
4. タグの相関関係（直接の子要素）でスタイルを設定する（p.30参照）
5. タグの相関関係（隣接する要素）でスタイルを設定する（p.33参照）
6. タグの属性によってスタイルを設定する（p.36参照）
7. 疑似クラス・疑似要素（p.39参照）

一見すると複雑ですが、このようにさまざまなセレクタを用意することで、柔軟にスタイルを指定することができるようになっています。

● 1..タグにスタイルを設定する

★ { ▲: △ }　　　1つのタグに設定

★,★…,★ { ▲: △ }　　　複数のタグに同一スタイルを設定

★••••••要素名
▲••••••プロパティ
△••••••値

タグにスタイルを設定するには、該当タグの要素名をセレクタにします。

複数のタグに対して同じスタイルを設定する場合には、要素名をそれぞれ「,」（カンマ）で区切って並べることでまとめてスタイルを指定することができます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>タグにスタイルを設定する</title>
<style type="text/css">
<!--
h1      { background-color: #cfc }
h1,h2   { color: rgb(0,128,0) }
-->
</style>
</head>

<body>
<h1>スタイルシートの設定方法</h1>
<p>
スタイルシートには幾つかの設定方法が用意されています。
</p>
<h2>外部スタイルシートを読み込む</h2>
<p>
```

```
スタイルの設定を行ったファイル（*.css）をHTML文書とは別に用意し、これを
&lt;link&gt;タグで読み込む設定方法です。
</p>
<h2>HTML文書内にまとめて設定する</h2>
<p>
HTML文書に設定したいスタイルを記述する方法です。&lt;style&gt;と&lt;/style&gt;タグ
の間でスタイルを定義し、これを&lt;head&gt;～&lt;/head&gt;内に配置します。
</p>
<h2>タグに直接スタイルを設定する</h2>
<p>
style属性を使用し、タグに直接スタイルを記述する方法です。
</p>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
	○	○	○	○	○	○	○

● 2..任意の範囲にスタイルを設定する

★.☆ { ▲: △ }　　<★ class="☆"> ～ </★>

★#◇{ ▲: △ }　　<★ id="◇"> ～ </★>

.☆ { ▲: △ }　　<◆ class="☆"> ～ </◆>

#◇ { ▲: △ }　　<◆ id="◇"> ～ </◆>

★,◆ •••要素名
☆••••••クラス名
◇••••••ID 名
▲••••••プロパティ
△••••••値

スタイルにクラスやIDという特定の識別子をふっておき、必要に応じてclass属性やid属性で適用させる方法です。複数の要素に対して同じスタイルを適用したり、逆に、比較的限定された任意の範囲にスタイルを適用するなど、前述（「タグにスタイルを設定する」）のような要素名だけではセレクタを限定できない場合に利用します。

クラスでは「.」（ピリオド）に続けてクラス名を、IDは「#」（シャープ）に続けてID名を指定します。クラス名とID名にはそれぞれ半角の英数字とハイフンを使って任意の名前を付けられますが、Netscape（Navigator）では1文字目を数字にすると、認識しないようですので、クラス名やID名の1文字目に数字をつけるのは避けるようにしましょう。なお、本書ではクラス名、ID名といった任意の名前は「***sample***」のように斜体で表現しています。

「.」や「#」の前に要素名を指定して「要素名.クラス名」「要素名#ID名」のようにすると指定された特定のタグでのみ有効になりますが、要素名が指定されていない場合（「.クラス名」「#ID名」）にはどのタグからでも呼び出すことができます。

なお、W3Cの定義によればIDは1つのHTML文書内で1度しか呼び出すことができないことになっています。現状のブラウザではIDであっても複数回呼び出すことができるようですが、これは正しい利用法ではありませんので注意しましょう。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>任意の範囲にスタイルを設定する</title>
<style type="text/css">
<!--
span.sample1        { color: blue }
strong#sample2      {
        color: white;
        background-color: #ff0099
}
.sample3            { font-weight: bold }
#sample4            { text-decoration: underline }
-->
</style>
</head>
<body>
<h1>スタイルシートとは</h1>
<p>
<span class="sample1">スタイルシートは、ひとことで表現するならば<strong
id="sample2">「Webページのレイアウトを定義する技術」</strong>ということがで
きます。</span><br>
スタイルシートを実現する<span id="sample4">CSS（Cascading Style Sheet）
</span>には次のものがあります。
</p>
<ul>
  <li><strong class="sample3">CSS1</strong>…1996年に勧告</li>
  <li><strong class="sample3">CSS2</strong>…1998年に勧告</li>
</ul>
</body>
</html>
```

▲任意の範囲にスタイルが適用できます

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
	○	○	○	○	○	○	○

※定義に反し、idを複数回呼び出せます

● 3..タグの相関関係でスタイルを設定する――子孫の要素に設定

★••••••親要素
☆••••••子孫の要素
▲••••••プロパティ
△••••••値

HTMLの木構造（p.4参照）を利用し、その相関関係によってスタイルを設定する方法です。

あるタグ（親要素）に含まれる特定のタグ（子孫である要素）に対してスタイルを設定する場合には、親要素のあとに半角スペースで区切って対象の要素を記述します。

サンプルでは、p要素に含まれるstrong要素に対してスタイルを設定しているため、span要素内にあるstrong要素にもスタイルが適用されています。一方li要素内に含まれるstrong要素には適用されません。

▲サンプルソースの木構造とスタイル適用の関係

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>タグの相関関係でスタイルを設定する</title>
<style type="text/css">
```

```
<!--
.blue          { color: blue }
p strong       {
        color: white;
        background-color: #ff0099
}
-->
</style>
</head>

<body>
<h1>スタイルシートとは</h1>
<p>
<span class="blue">スタイルシートは、ひとことで表現するならば<strong>「Webページのレイアウトを定義する技術」</strong>ということができます。</span>
スタイルシートを実現するCSS（Cascading Style Sheet）には次のものがあります。
</p>
<ul>
  <li><strong>CSS1</strong>…1996年に勧告</li>
  <li><strong>CSS2</strong>…1998年に勧告</li>
</ul>
</body>
</html>
```

▲<P>タグ内の<strong>にのみスタイルを適用しているため、その他の<strong>にはスタイルは適用されません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
	○	○	○	○	○	○	○

● 4..タグの相関関係でスタイルを設定する――直接の子要素に設定

★・・・・・・親要素
☆・・・・・・子要素
▲・・・・・・プロパティ
△・・・・・・値

HTMLの木構造（p.4）を利用し、その相関関係によってスタイルを設定する方法です。

あるタグ（親要素）の直下のタグ（直接の子要素）に対してスタイルを設定する場合には、親要素と対象の子要素を「>」を挟んで記述します。

サンプルでは、p要素の直接の子要素であるstrong要素に対してスタイルを設定しています。そのため、span要素やli要素内に含まれるstrong要素にはstrong要素の効果のみで、スタイルは適用されません。

▲サンプルソースの木構造とスタイル適用の関係

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>タグの相関関係でスタイルを設定する</title>
<style type="text/css">
<!--
.blue           { color: blue }
p > strong      {
        color: white;
        background-color: #ff0099
}
-->
</style>
</head>

<body>
<h1>スタイルシートとは</h1>
<p>
<span class="blue">スタイルシートは、ひとことで表現するならば<strong>「Web
ページのレイアウトを定義する技術」</strong>ということができます。</span>
スタイルシートを実現する<strong>CSS（Cascading Style Sheet）</strong>には次のも
のがあります。
</p>
<ul>
  <li><strong>CSS1</strong>…1996年に勧告</li>
  <li><strong>CSS2</strong>…1998年に勧告</li>
</ul>
</body>
</html>
```

▲Internet Explorerは対応していません

▲<p>タグ直下の<strong>タグにのみスタイルが適用されます

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
	×	×	×	×	×	×	○

※Macintosh版IE5は対応しています

● 5..タグの相関関係でスタイルを設定する——隣接する要素に設定

★・・・・・・・前要素
☆・・・・・・・後要素
▲・・・・・・・プロパティ
△・・・・・・・値

HTMLの木構造（p.4参照）を利用し、その相関関係によってスタイルを設定する方法ですが、前述の方法のような親子関係ではなく、タグの前後関係によってスタイルを設定します。

この方法では、同じ親をもち、要素（前要素）の次に出現する特定の要素（後要素）にのみスタイルを設定することができます。

サンプルでは、ul要素の直後のp要素にのみスタイルが適用され、h1要素の後にくるp要素には適用されません。

▲サンプルソースの構造とスタイル適用の関係

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>タグの相関関係でスタイルを設定する</title>
<style type="text/css">
<!--
ul + p {
        margin:30pt 50pt;
        border-color: gray;
        border-style: dotted
}
-->
</style>
</head>

<body>
<h1>スタイルシートとは</h1>
<p>
スタイルシートを実現する<strong>CSS（Cascading Style Sheet）</strong>には次のも
のがあります。
</p>
<ul>
  <li>CSS1…1996年に勧告</li>
  <li>CSS2…1998年に勧告</li>
</ul>
<p>
W3Cは1996年12月にスタイルシート言語の一つであるCSS1（Cascading Style Sheets
Level 1）を勧告し、Internet Explorer3.0とNetscape Navigator 4.0がこの技術を導入しはじ
めました。CSS1ではHTMLで可能だったデザインのほとんどを扱えるようになっています。
その後1998年5月12日には次の規格であるCSS2が出されました。</p>
</body>
</html>
```

▲Internet Explorerは対応していません

▲<ul>タグに続く<P>タグにのみスタイルが適用されます

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
	×	×	×	×	×	×	○

※Macintosh版IE5は対応しています

● 6..タグの属性によってスタイルを設定する

★[◆] { ▲: △ }　　属性を利用

★[◆="◇"] { ▲: △ }　　属性と値を利用

★……要素名
◆……要素の属性
◇……属性の値
▲……プロパティ
△……値

タグの属性を利用してスタイルを指定する方法です。

★[◆]の形式では属性の値に関係なく、指定された属性を持つ要素にスタイルを設定することができます。一方★[◆="◇"]の形式では、指定された属性と値を持つ要素にのみスタイルを設定することができます。

サンプルではhref属性を持つa要素すべてのアンダーラインをなくし、背景色に水色を指定しています。また、topへのリンクを持つa要素は文字サイズと背景色を変えています。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>タグの属性によってスタイルを設定する</title>
<style type="text/css">
<!--
a[href]           {
        text-decoration: none;
        background-color: aqua
}
a[href="#top"]    {
        font-size: xx-small;
        background-color: yellow
}
p.title           {
```

```
        font-size: large;
        font-weight: bold
}
-->
</style>
</head>

<body>
<p class="title">スタイルシートの設定方法</p>
<ul>
  <li><a href="#c1">外部スタイルシートを読み込む</a></li>
  <li><a href="#c2">HTML文書内にまとめて設定する</a></li>
  <li><a href="#c3">タグに直接スタイルを設定する</a></li>
</ul>
<hr>
<p class="title"><a name="c1">外部スタイルシートを読み込む</a></p>
<p>
スタイルの設定を行ったファイルをHTML文書とは別に用意し、これを&lt;link&gt;タグで
読み込む設定方法です。
<a href="#top">目次へ戻る</a>
</p>
<p class="title"><a name="c2">HTML文書内にまとめて設定する</a></p>
<p>
HTML文書に設定したいスタイルを記述する方法です。&lt;style&gt;と&lt;/style&gt;タグ
の間でスタイルを定義し、これを&lt;head&gt;～&lt;/head&gt;内に配置します。
<a href="#top">目次へ戻る</a>
</p>
</body>
</html>
```

▲Internet Explorerは対応していません

IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
×	×	×	×	×	×	○

疑似クラス・疑似要素

★:☆ { ▲: △ }

★••••••要素名
☆••••••疑似クラス、疑似要素
▲••••••プロパティ
△••••••値

セレクタには疑似クラス、疑似要素というものもあります。

疑似クラス

要素名や属性などで分類できない状態に対してスタイルを設定するために、疑似クラスというものがあり、次のような疑似クラスが定義されています。要素名に続けて記述してください。

なお、:hoverをリンクに設定する場合（a:hover）は、a:link、a:visitid、a:hover、a:activeの順番に記述する必要があります。

:link	まだ見ていない（キャッシュされていない）ページへのリンク
:visited	すでに見た（キャッシュされている）ページへのリンク
:hover	マウスカーソルが要素と重なっているとき（まだアクティブではないとき）
:active	リンク部分を選択した瞬間（クリックなど）
:focus	対象がクリックされたとき
:lang	（言語コード）スタイルを適用させる言語

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>疑似クラス</title>
<style type="text/css">
<!--
a:link          { color: #0000ff }
a:visited       { color: #080 }
```

```
a:hover          { color: rgb(128,000,128) }
a:active         { color: fuchsia }
}
-->
</style>
</head>

<body>
<p style="font-size: large; font-weight: bold;">リンクにスタイルを設定</p>
<ul>
  <li><a href="sample1.html">まだ見ていないページへのリンクは青</a></li>
  <li><a href="sample2.html">すでに見たページへのリンクは緑</a></li>
  <li><a href="sample3.html">マウスカーソルが要素と重なっているときは紫</a></li>
  <li><a href="sample4.html">選択されたリンクは紅紫</a></li>
</ul>
</body>
</html>
```

▲Internet Explorerの場合

疑似クラス	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
:link	○	○	○	○	○	○	○
:visited	○	○	○	○	×	○	○
:active	○	○	○	○	×	×	○
:hover	○	○	○	○	×	×	○
:focus	×	×	×	×	×	×	○
:lang	×	×	×	×	×	×	×

※適用するタグによっても効果が変わります
※Macintosh版IE5はfocusにも対応しています

疑似要素

木構造を構成する要素などでは指定できない部分に対してスタイルを設定するために、次のような疑似要素が定義されています。要素名に続けて記述してください（p.4参照）。

:first-line	要素の表示上の最初の１行（ブロックレベル要素にのみ指定可能）
:first-letter	要素の最初の１文字（一部のブロックレベル要素にのみ指定可能）
:before	要素の直前に生成追加される内容（content プロパティと併用）
:after	要素の直後に生成追加される内容（content プロパティと併用）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>疑似要素</title>
<style type="text/css">
<!--
div:first-line  { color: #ff0099 }     /*1行目を赤紫に*/
p:first-letter  {                      /*1文字目を大きく*/
        font-size: 300%;
        font-weight: bold;
        color: green;
        float: left
}
div:before      {                      /*要素の前に「Check!」の文字*/
        content: "Check!";
        font-size: x-small;
        color: red
}
p:after         {                      /*要素の後に「OK?」の文字*/
        content:"OK?";
        font-size: x-small;
        color: gray
}
```

```
-->
</style>
</head>

<body>
<div>
スタイルシートは、ひとことで表現するならば「Webページのレイアウトを定義する技術」ということができます。文書の論理構造に関する指定と体裁に関わる指定とを分離させ、本来HTMLの機能ではない体裁の制御については別の方法を導入しようという姿勢のもとに生み出されました。
</div>
<p>
W3Cは1996年12月にスタイルシート言語の一つであるCSS1（Cascading Style Sheets Level 1）を勧告し、Internet Explorer3.0とNetscape Navigator 4.0がこの技術を導入しはじめました。CSS1ではHTMLで可能だったデザインのほとんどを扱えるようになっています。その後1998年5月12日には次の規格であるCSS2が出されました。
</p>

</body>
</html>
```

▲Internet Explorerは：first-letterと：first-lineのみ対応しています

▲Netscape 6.2では：beforeと：afterにも対応しています

疑似要素	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
:first-letter	×	×	○	○	×	×	○
:first-line	×	×	○	○	×	×	○
:before	×	×	×	×	×	×	○
:after	×	×	×	×	×	×	○

※適用するタグによっても効果が変わります

※Macintosh版IE5はfirst-letter、first-lineにも対応しています

9 STYLESHEET BASIC

スタイルの継承

プロパティには子要素によって値が継承されるものがあります。たとえば、

body { color : blue } /* ページ内のテキストの色を青に指定 */

このような指定があった場合、特に文字色の指定されていない子要素は、body 要素の指定値を継承して青で表示することになります（ただし、継承される値が相対値で指定されていた場合は、まず相対値を絶対値として算出してから継承します)。

値が継承されないプロパティには、背景画像やマージン、パディングの指定などがあります。たとえば、

div { margin: 5pt } /*div 要素のマージンを 5pt に指定 */

という指定をした場合、div 要素のマージンは 5pt になりますが、マージンの値は継承されないため、この要素内の子要素や子孫の要素にマージンを設定したい場合には、新たに該当の要素に対してスタイルを指定する必要があります。

プロパティの値が継承するかどうかについては、付録の「適用・デフォルト・継承一覧」を参照してください。

また、各プロパティには「inherit」という値を指定することで、通常は値を継承しない要素でも、強制的に親要素の値を継承させることができます。

10 STYLESHEET BASIC

ボックスの概念

スタイルシートでは、すべての要素はボックスと呼ばれる四角い領域を持つと考え、この領域や領域を囲む枠線に対して大きさや色・位置の指定をすることでスタイルを変更します。ボックスは内容領域・マージン・パディング・枠線の４つの部分から構成されており、図のような構造になっています。

こうした概念を用いることで、枠線・枠線と要素の余白・枠線の外側の余白…上下左右といったように細かくスタイルが設定できます。

内容領域

テキストや画像など、要素の内容が表示される領域です。widthプロパティやheightプロパティでサイズを指定することができます。

パディング

要素の内容が表示される部分と枠線との間の余白領域です。要素のbackground-colorプロパティで指定した背景色はこの部分にも適用されます。

枠線

要素の周りに表示される枠で、パディングの外側に設定されます。

マージン

枠線の外側に設定される余白領域です。要素のbackground-colorプロパティで指定した背景色はこの部分には適用されず、背景は常に透明になります。そのため、親要素に背景が設定されている場合には、その背景が透けて見えることになります。

背景色・背景画像

内容領域とパディング領域（枠線の内側）に表示される色や画像です。

11 STYLESHEET BASIC

スタイルシートにおける単位

スタイルシートで長さを指定するには、相対単位で指定する方法と絶対単位で指定する方法の2通りあり、それぞれ以下の単位があります。

相対単位

単位	意　味	指定例
em	その要素のfont-sizeの値を1とする	h1 { margin: 0.5em }
ex	その要素のフォントのx-height（=小文字の「x」の高さ）を1とする	h1 { margin: 1ex }
px	コンピュータ画面上の1ピクセルを1とする	p { font-size: 12px }
%	多くの場合は親要素の一部分を基準とした割合（属性によって異なる）	p { font-size: 120% }

絶対単位

単位	意　味	指定例
in	インチ（1in=2.54cm）	h1 { margin: 0.5in }
cm	センチメートル	h2 { line-height: 3cm }
mm	ミリメートル	h3 { word-spacing: 4mm }
pt	ポイント（1pt=1/72in）	h4 { font-size: 12pt }
pc	パイカ（1pc=12pt）	h4 { font-size: 1pc }

12 STYLESHEET BASIC

スタイルシートにおける色

スタイルシートで色を指定するには、RGB値を用いる方法と、キーワードを用いる方法とがあり、それぞれ次のように指定します（具体例はいずれも赤を指定する場合です）。

なお、ブラウザによっては対応していない指定方法や色もあります。注意して使用するようにしましょう。

RGB値による指定

#rrggbb

「#」につづけて赤（r）、緑（g）、青（b）のそれぞれの値を00～ffの16進数で2桁ずつ、計6桁で指定します。

例：**#ff0000**

#rgb

「#」につづけて赤（r）、緑（g）、青（b）のそれぞれの値を0～fの16進数で1桁ずつ、計3桁で指定します。この方法ではrgb各桁を2つ繰り返して並べた6桁の形式（#rrggbb）に変換されてから色が表現されます。たとえば「#fb0」という値は「#ffbb00」という値に変換されることになります。

例：**#f00**

rgb(n,n,n)

rgbにつづく「()」の中に赤（r）、緑（g）、青（b）のそれぞれの値を「,」で区切って10進数の整数で指定します。

例：**rgb(255,0,0)**

rgb(n%,n%,n%)

rgbにつづく「()」の中に赤（r）、緑（g）、青（b）のそれぞれの値を「,」で区切ってパーセントで指定します。

例：**rgb(100%,0%,0%)**

キーワードによる指定

色名による指定

色名で直接指定します。大文字と小文字は区別されません。HTML4.01では基本的な16色が定義されています。

基本的な16色については下表を、その他の色名についてはp.342のカラーチャートを参照してください。

red	#ff0000	navy	#000080	green	#008000	black	#000000
fuchsia	#ff00ff	blue	#0000ff	lime	#00ff00	gray	#808080
purple	#800080	aqua	#00ffff	olive	#808000	silver	#c0c0c0
maroon	#800000	teal	#008080	yellow	#ffff00	white	#ffffff

システムカラーによる指定

スタイルシートでは、WindowsやMacOSが保持しているシステム情報を呼び出すことができます。このシステムカラーを使うと、ページを見る人のOSの環境に合わせて使用色を決めることができます。

システムカラーには以下のキーワードがあります。

activeborder	アクティブなウィンドウの枠の色
activecaption	アクティブなウィンドウのタイトルバーの色
appworkspace	アプリケーション内のウィンドウの背景色
background	デスクトップの背景色
buttonface	立体的ボタンの表面の色
buttonhighlight	立体的なボタンの光のあたっている面の色
buttonshadow	立体的なボタンの影になってる面の色
buttontext	立体的なボタンのテキストの色
captiontext	タイトルバーのテキストの色
graytext	選択できないテキストの色
highlight	選択している状態の色
highlighttext	選択しているテキストの色
inactiveborder	アクティブでないウィンドウの枠の色
inactivecaption	アクティブでないウィンドウのタイトルバーの色
inactivecaptiontext	アクティブでないウィンドウのタイトルバーのテキストの色
infobackground	ツールチップの背景色
infotext	ツールチップのテキストの色
menu	メニューの背景色

menutext　メニューのテキストの色
scrollbar　スクロールバーの色
threeddarkshadow　立体的に表示される部分の暗い影の色
threedface　立体的に表示される部分の表面の色
threedhighlight　立体的に表示される部分の光のあたっている面の色
threedlightshadow　立体的に表示される部分の明るい影の色
threedshadow　立体的に表示される部分の影の色
window　ウィンドウの背景色
windowframe　ウィンドウのフレームの色
windowtext　ウィンドウのテキストの色

▲システムカラーによる指定は、OSの違いなどマシンの環境によって異なります

transparentの指定

プロパティによってはtransparent（透明）を指定できるものもあります。transparentを指定するとその領域は透明になり、結果としてその要素が含まれる親ボックスの（つまり下の）内容や背景・背景画像などが透けて見えるようになります。

13 STYLESHEET BASIC

URLの書き方

スタイルシートでURLやファイルの位置を指定する場合には、「url()」を使用します。URLは引用符（「" "」「' '」）でくくることもでき、またURLの前後には空白スペースを入れることもできます。

例：　**url(http://www.ank.co.jp/)**
　　　url("../books/css.html")

URIとURL

URLを指定するには、あるファイルの位置を一番もとになる位置から階層構造を順番にたどって指定する絶対URLと、ファイル同士の位置を、基準となるファイルから見てどこにあるか、階層の上下を表すことで指定する相対URLの2通りの方法があります。主に、絶対URLは他のサイトにあるファイルを指定する場合、相対URLは自分のサイト内にあるファイルを指定する場合に使用します。もちろん、自分のサイト内のファイルに絶対URLを指定することも可能ですが、オンラインの状態で使用することや、フォルダごと移動させたい場合のことを考慮すると、相対URLで記述しておいたほうが便利でしょう。

なお、HTML4.0からは「URL」に代わってより広義な「URI（Universal Resource Identifers)」という用語が使用されるようになっています。URLと同様にHTML文書・画像・ビデオクリップ・プログラムなどを指定できますが、URLはURIのサブセットでURIのほうが上位の概念です。

本書では、読者にとって馴染みの深いと思われるURLを使用していますが、URIとするのが正しい表現ですので、ぜひ覚えてください。

14 STYLESHEET BASIC

DTDとブラウザでの表示

従来のInternet ExplorerやNetscapeでは、HTML文書の冒頭に記述する文書型宣言（<!DOCTYPE>）の有無や、文書型宣言の後半のURL部分を省略するかどうかといった表記の違いが、コンテンツの表示に直接影響を与えることはありませんでした。しかし、Windows版のInternet Explorer 6、Mac版Internet Explorer 5.x、Netscape 6以降では、以下の2通りの表示モードが用意され、文書型宣言の書き方でこれらの表示モードが切り替わる仕組みになっています。

標準準拠モード

W3Cの標準仕様に従って正しく表示をする

互換モード

旧バージョンとの互換性を考慮し従来のブラウザと同様の表示をする

表示の違いが現れるのは主にスタイルシートを利用した時ですが、HTMLだけで文書を作成した場合にも多少の影響が出ますので注意してください。

HTML4.01のDOCTYPE宣言の記述方法と、表示モードとの関係は次のようになっています。

DTD バージョン	記述方法	IE6	N6.2	Mac版 IE5
-	記述なし	互換	互換	互換
HTML4.01 Strict	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">	標準	標準	標準
	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">	標準	標準	互換
HTML4.01 Transitional	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">	標準	標準	標準
	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">	互換	互換	互換

たとえば、文書型宣言の記述方法以外はまったく同じ内容を持つ以下のサンプルを、ブラウザに表示させると次のような違いが現れます。

URLを省略せずにDTDを宣言：標準準拠モード

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<title>標準準拠モードのテスト</title>
<style type="text/css">
<!--
div     {
        width: 300px;
        padding: 20px;
        border: dotted 3px #880000
}
-->
</style>
</head>

<body>
<h1>標準準拠モード</h1>
<div>
Webページを記述するHTMLは、文書の論理的な構造を示す言語です。たとえば、見出しが
あり、本文の中に段落やリストがあり……といったように、文書がどのような要素で構成さ
れているのか、またある特定の部分が文書全体の中でどのような意味をもっているのかを、
コンピュータに知らせるための言語なのです。表現方法を指定するための言語ではありませ
ん。しかし実際は、Webの発展にともない、色やフォントサイズの指定、レイアウトのた
めのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するようになっていきまし
た。W3Cではこの状況を改めるため、構造に関する指定と体裁に関わる指定とを分離させ、
HTMLの機能ではない体裁の制御については別の方法を導入しようと考えるようになりまし
た。こうした姿勢のもとに生み出されたのがスタイルシートの概念です。
</div>
</body>
</html>
```

URLを省略してDTDを宣言：互換モード

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<title>互換モードのテスト</title>
<style type="text/css">
<!--
div     {
        width: 300px;
        padding: 20px;
        border: dotted 3px #880000
}
-->
</style>
</head>

<body>
<h1>互換モード</h1>
<div>
Webページを記述するHTMLは、文書の論理的な構造を示す言語です。たとえば、見出しがあり、本文の中に段落やリストがあり……といったように、文書がどのような要素で構成されているのか、またある特定の部分が文書全体の中でどのような意味をもっているのかを、コンピュータに知らせるための言語なのです。表現方法を指定するための言語ではありません。しかし実際は、Webの発展にともない、色やフォントサイズの指定、レイアウトのためのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するようになっていきました。W3Cではこの状況を改めるため、構造に関する指定と体裁に関わる指定とを分離させ、HTMLの機能ではない体裁の制御については別の方法を導入しようと考えるようになりました。こうした姿勢のもとに生み出されたのがスタイルシートの概念です。
</div>
</body>
</html>
```

▲標準準拠モード。内容領域のサイズがwidthによって設定されます

▲互換モード。内容領域にパディングと枠線を加えた領域のサイズがwidthによって設定されます

テキスト/TEXT
フォント/FONT
背景/BACKGROUND
ボックス/BOX
配置/POSITIONING
リスト/LIST
テーブル/TABLE
フィルタ/FILTER
その他/OTHER

第2部

スタイルシートリファレンス

STYLESHEET REFERENCE

1 TEXT

文字色を指定したい

color: ★

★……RGB値
　　キーワード

文字色を指定します。

色の指定には、RGBの数値で指定する方法と、キーワードで指定する方法とがあります。色の詳しい指定方法についてはp.47を参照してください。

なお、色の指定がされていない文字には、ユーザーのブラウザの設定にしたがった色が表示されます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>文字色を指定したい</title>
<style type="text/css">
<!--
body        { background-color: black }
.sample1    { color: #ff0000 }
.sample2    { color: #ff00ff }
.sample3    { color: #008800 }
.sample4    { color: #fff }
.sample5    { color: #0f0 }
.sample6    { color: #00f }
.sample7    { color: rgb(128,0,0) }
.sample8    { color: rgb(255,255,0) }
.sample9    { color: rgb(0,128,128) }
```

```
.sample10      { color: rgb(75%,75%,75%) }
.sample11      { color: rgb(50%,0%,50%) }
.sample12      { color: rgb(0%,0%,50%) }
.sample13      { color: gray }
.sample14      { color: olive }
.sample15      { color: aqua }
-->
</style>
</head>
<body>
<p>
<span class="sample1">テ</span>
<span class="sample2">キ</span>
<span class="sample3">ス</span>
<span class="sample4">ト</span>
<span class="sample5">の</span>
<span class="sample6">色</span>
<span class="sample7">を</span>
<span class="sample8">変</span>
<span class="sample9">更</span>
<span class="sample10">し</span>
<span class="sample11">て</span>
<span class="sample12">み</span>
<span class="sample13">ま</span>
<span class="sample14">す</span>
<span class="sample15">。</span>
</p>
</body>
</html>
```

テキスト

文字色を指定するHTMLタグをCSSに改める

HTMLタグでテキストの色を指定するには、次のように<body>タグまたはタグの属性を利用します。

<body text="★">～</body>	標準の文字色
<body link="★">～</body>	まだ見ていないページへのリンク部分の色
<body alink="★">～</body>	リンク部分を選択した瞬間の色
<body vlink="★">～</body>	すでに見たページへのリンク部分の色
～	部分的なテキストの色指定

★――色の指定値

HTMLタグの文字色を指定する属性はすべてDeprecated（推奨しない）とされており、代わりにスタイルシートで指定することが推奨されています。同様の効果をスタイルシートで表現すると以下のようになります（疑似クラスについてはp.39参照）。

body	{ color: ★ }	標準の文字色
a:link	{ color: ★ }	まだ見ていないページへのリンク部分の色
a:active	{ color: ★ }	リンク部分を選択した瞬間の色
a:visited	{ color: ★ }	すでに見たページへのリンク部分の色
☆	{ color: ★ }	部分的な文字色の指定（☆――セレクタ）

★――色の指定値

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
色名	○	○	○	○	○	○	○
システムカラー	○	○	○	○	×	×	○
#rgb	○	○	○	○	○	○	○
#rrggbb	○	○	○	○	○	○	○
rgb(n,n,n)	○	○	○	○	○	○	○
rgb(n%,n%,n%)	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

背景色を指定したい……………………………p.136
背景画像を指定したい…………………………p.140

2 TEXT

文字を装飾したい

text-decoration: ★

★……キーワード

指定した要素に含まれる文字に対して、上線、下線、取り消し線、点滅といった装飾を設定します。

値には以下のキーワードがあります。

overline	上線
underline	下線
line-through	取り消し線
blink	点滅
none	装飾なし（デフォルト）

それぞれを半角スペースで区切って並べれば、複数の値を指定することもできます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>文字を装飾したい</title>
<style type="text/css">
<!--
p.sample1     { text-decoration: none }
p.sample2     {
        text-decoration: overline underline;
        color: red
}
p.sample3     {
        text-decoration: line-through;
```

```
        color: blue
}
p.sample4    {
        text-decoration: blink;
        color: green
}
-->
</style>
</head>
<body>
<p class="sample1">text-decoration: none 装飾なし（デフォルト）</p>
<p class="sample2">text-decoration: overline underline で上線と下線を指定 </p>
<p class="sample3">text-decoration: line-through で取り消し線を指定 </p>
<p class="sample4">text-decoration: blink で点滅を指定 </p>
</body>
</html>
```

▲Internet Explorer は blink のみ対応していません

▲Netscape 6.2 では blink に対応しなくなりました

文字装飾を指定するHTMLタグをCSSに改める

HTMLタグで文字を装飾するには、次のようなタグを利用します。

<u>～</u>	下線
<strike>～</strike>	取り消し線
<s>～</s>	取り消し線
<blink>～</blink>	点滅

<strike>、<s>、<u>タグは、Deprecated（推奨しない）とされており、こうした文字の装飾はスタイルシートで指定することが推奨されています。同様の効果をスタイルシートで表現すると以下のようになります。

★ { text-decoration: underline }	下線
★ { text-decoration: line-though }	取り消し線
★ { text-decoration: line-though }	取り消し線
★ { text-decoration: blink }	点滅

★──セレクタ

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
none	○	○	○	○	○	○	○
overline	○	○	○	○	×	×	○
underline	○	○	○	○	○	○	○
line-through	○	○	○	○	○	○	○
blink	×	×	×	×	○	○	×

※適用するセレクタによっても効果が変わります

縦書きの下線（傍線）位置を指定したい…………p.102
フォントを斜体にしたい…………………………p.126

大文字・小文字に置換したい

text-transform: ★

★••••••キーワード

英文の大文字・小文字の表記方法を指定します。英文をすべて大文字（あるいは小文字）で表示したり、単語の一文字目を大文字で表示したりすることができます。

値には以下のキーワードがあります。

capitalize	各単語の１文字目を大文字に置換
uppercase	すべての文字を大文字に置換
lowercase	すべての文字を小文字に置換
none	入力されたままで表示（デフォルト）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>大文字・小文字に置換したい</title>
<style type="text/css">
<!--
div          {
        margin: 20pt auto;
        text-transform: none
 }
.sample1     {
        text-transform: capitalize;
        color: red
}
```

```
.sample2	{
	text-transform: uppercase;
	color: blue
}
.sample3	{
	text-transform: lowercase;
	color: green
}
-->
</style>
</head>
<body>
<div>
text-transform: capitalize<br>
<span class="sample1">the text-transform property controls capitalization effects of an element's text. </span>
</div>
<div>
text-transform: uppercase<br>
<span class="sample2">the text-transform property controls capitalization effects of an element's text. </span>
</div>
<div>
text-transform: lowercase<br>
<span class="sample3">THE TEXT-TRANSFORM PROPERTY CONTROLS CAPITALIZATION EFFECTS OF AN ELEMENT'S TEXT.</span>
</div>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
capitalize	○	○	○	○	×	×	○
uppercase	○	○	○	○	×	×	○
lowercase	○	○	○	○	×	×	○
none	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

文字にスモールキャピタルを指定したい……….p.128

4 TEXT

行の高さを設定したい

line-height: ★

★……キーワード
サイズを表す数値＋単位
割合を表す数値
パーセントを表す数値＋%

行の高さ（行送り）を設定します。
値には次のような指定方法があります。

キーワード

normal　ブラウザが自動的に行の高さを設定（デフォルト）

サイズを表す数値＋単位

ボックス領域の高さ（1行の上端から下端までの幅）を、数値に単位をつけて設定します。フォントサイズが12ptのときに12ptや1emを設定すると行間が0になり、またフォントサイズが12ptの時に10ptを設定するとテキストが次の行にはみ出してしまいますので注意してください。単位についてはp.46を参照してください。

割合を表す数値（単位なし）

その要素のフォントサイズ（その要素に設定されているサイズや親要素から継承してきたサイズ）に対する行の高さの割合を、単位なしの数値で設定します。たとえばフォントサイズが10pxのときに1.5を設定すると、行の高さは15pxとなります。

パーセントを表す数値＋%

その要素のフォントサイズ（その要素に設定されているサイズや親要素から継承してきたサイズ）に対する行の高さの割合を、パーセンテージで設定します。

次の指定はいずれも同じ結果になります。

```
div { font-size: 10pt; line-height: 1.2em }
div { font-size: 10pt; line-height: 1.2 }
div { font-size: 10pt; line-height: 120% }
```

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>行の高さを設定したい</title>
<style type="text/css">
<!--
div             {
        font-size: 16pt;
        margin: 20px auto 20px 20px;
        width: 500px;
        padding: 20px;
        background-color: #ff9966
}
.sample1        { line-height: normal }
.sample2        { line-height: 2 }
-->
</style>
</head>
<body>
<div class="sample1">
スタイルシートは、ひとことで表現するならば「Webページのレイアウトを定義する技術」
ということができます。文書の論理構造に関する指定と体裁に関わる指定とを分離させ、本
来HTMLの機能ではない体裁の制御については別の方法を導入しようという姿勢のもとに生
み出されました。
</div>
<div class="sample2">
スタイルシートは、ひとことで表現するならば「Webページのレイアウトを定義する技術」
ということができます。文書の論理構造に関する指定と体裁に関わる指定とを分離させ、…
…（中略）……姿勢のもとに生み出されました。
</div>
</body>
</html>
```

▲値を2に設定すると、行の高さはフォントサイズの2倍（32pt）になります

▲値を2に設定すると、行の高さはフォントサイズの2倍（32pt）になります

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○
比率	○	○	○	○	○	○	○
normal	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照
文字の垂直位置を指定したい……p.72
文字をグリッドにおさめたい……p.108
フォントサイズを指定したい……p.117

行揃えを指定したい

text-align: ★

★••••••キーワード

行揃えを指定します。

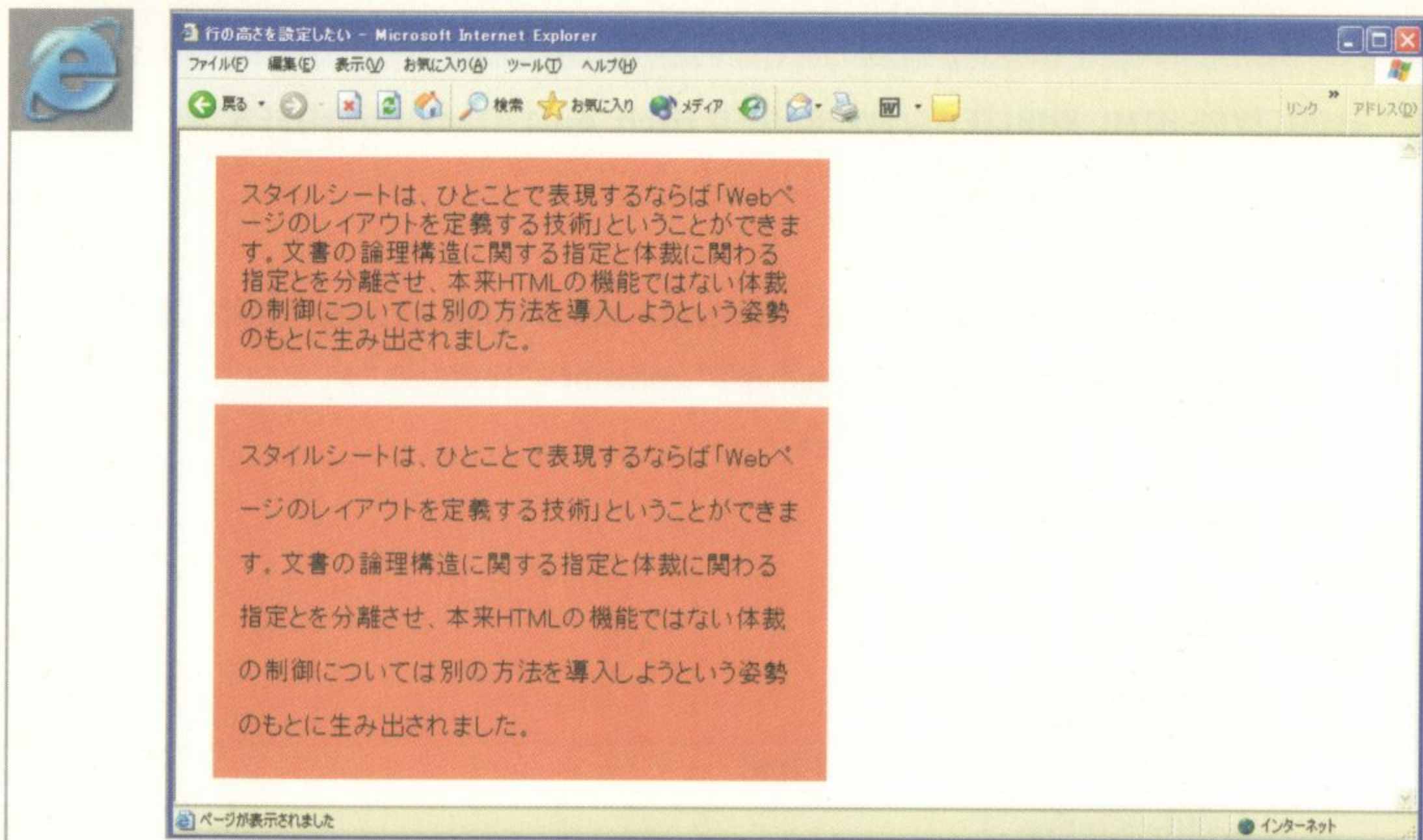
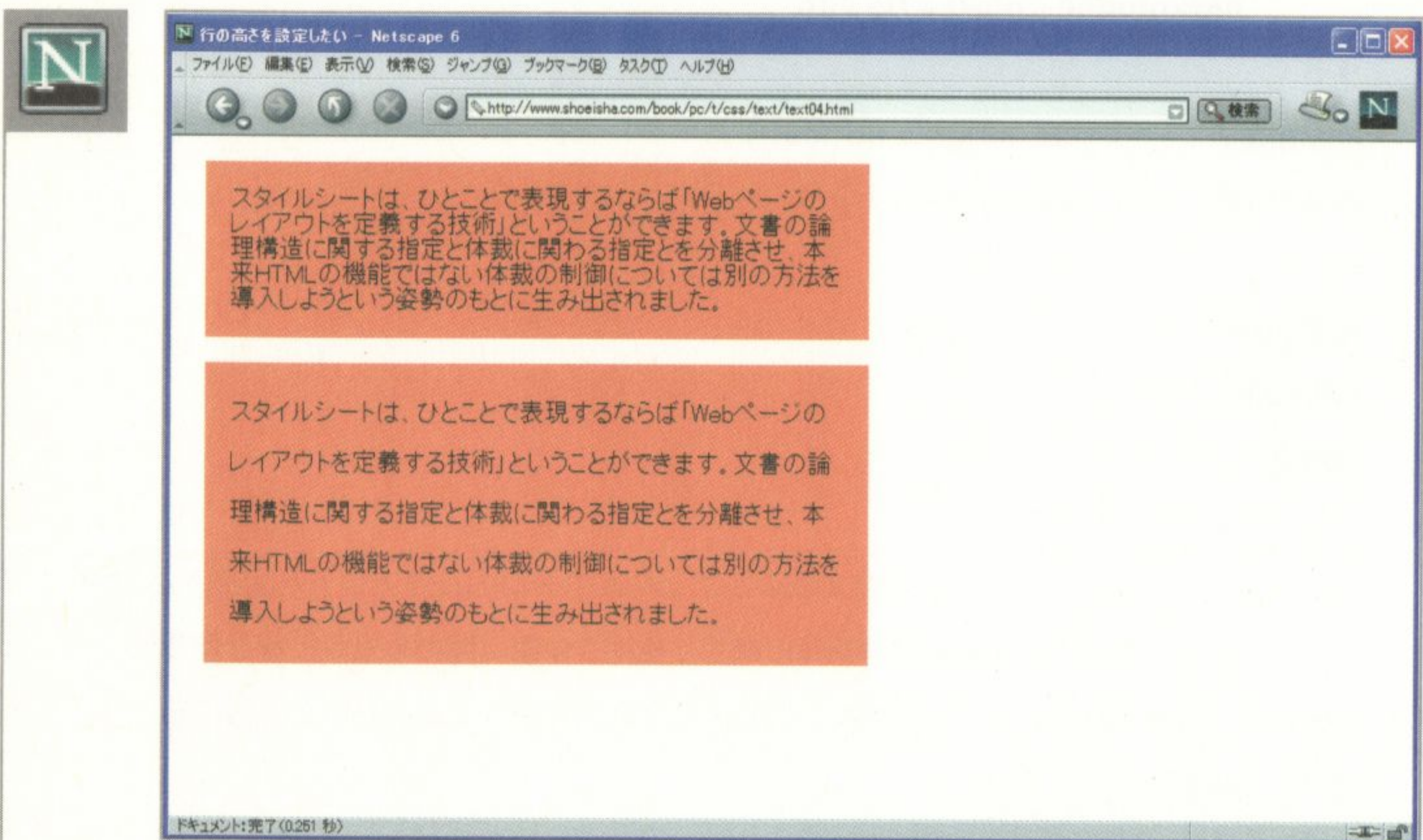
値には以下のキーワードがあります。

left	左寄せ
right	右寄せ
center	センタリング
justify	均等割付

justifyは均等割付を行う値ですが、実際はtext-alignプロパティ単体では動作しません。均等割付についてはtext-jusifyプロパティ（p.76）を参照してください。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>行揃えを指定したい</title>
<style type="text/css">
<!--
.sample1	{
	text-align: left;
	color: red
}
.sample2	{
	text-align: right;
	color: green
```

```
}
.sample3        {
        text-align: center;
        color: blue
}
-->
</style>
</head>
<body>
<div class="sample1">text-align: left で左寄せを指定。</div>
<div class="sample2">text-align: right で右寄せを指定。</div>
<div class="sample3">text-align: center でセンタリングを指定。</div>
<br>
<table border="1" width="100%">
   <tr><th class="sample1"> 見出し左寄せ </th>
   <th class="sample2"> 見出し右寄せ </th></tr>
   <tr class="sample3"><td> 行センタリング </td>
   <td> 行センタリング </td></tr>
   <tr><td class="sample2"> セル右寄せ </td>
   <td class="sample1"> セル左寄せ </td></tr>
</table>
</body>
</html>
```

行揃えを指定したい - Netscape 6

ファイル(F) 編集(E) 表示(V) 検索(S) ジャンプ(G) ブックマーク(B) タスク(T) ヘルプ(H)

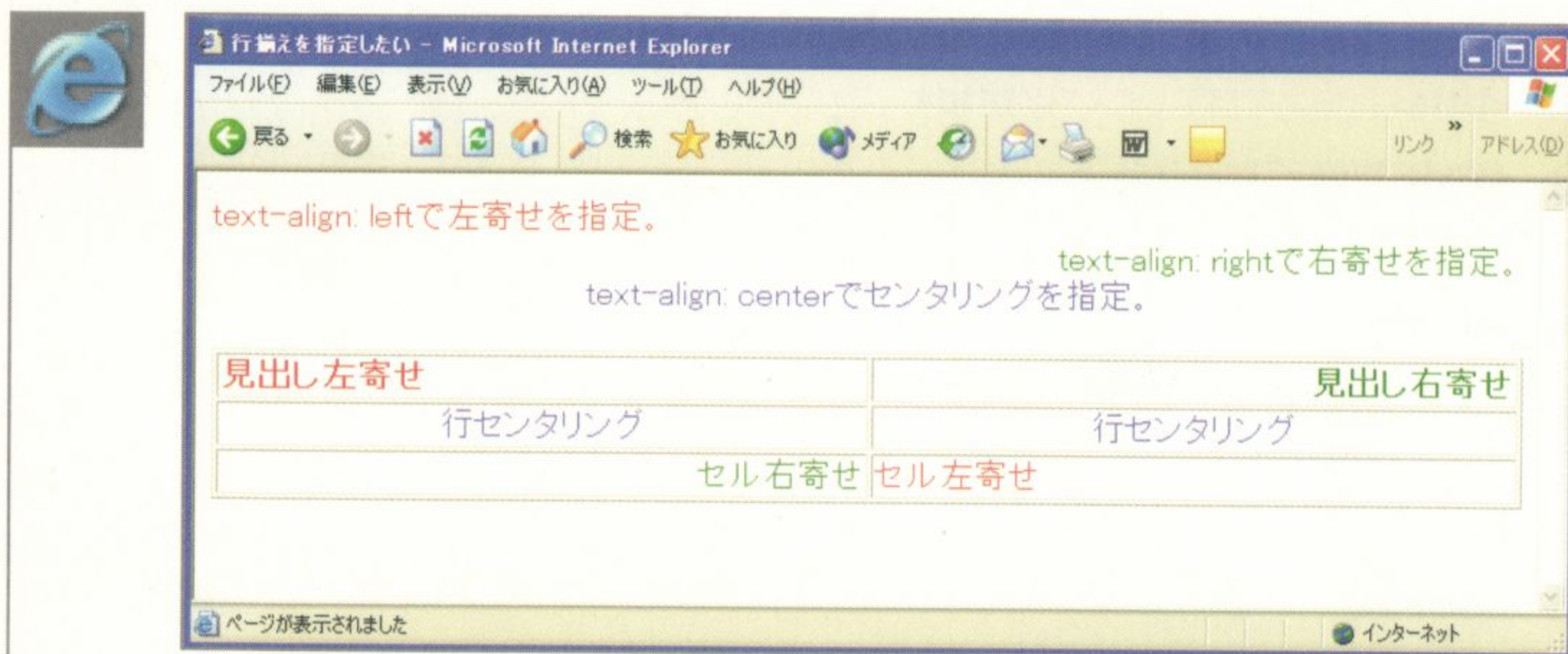
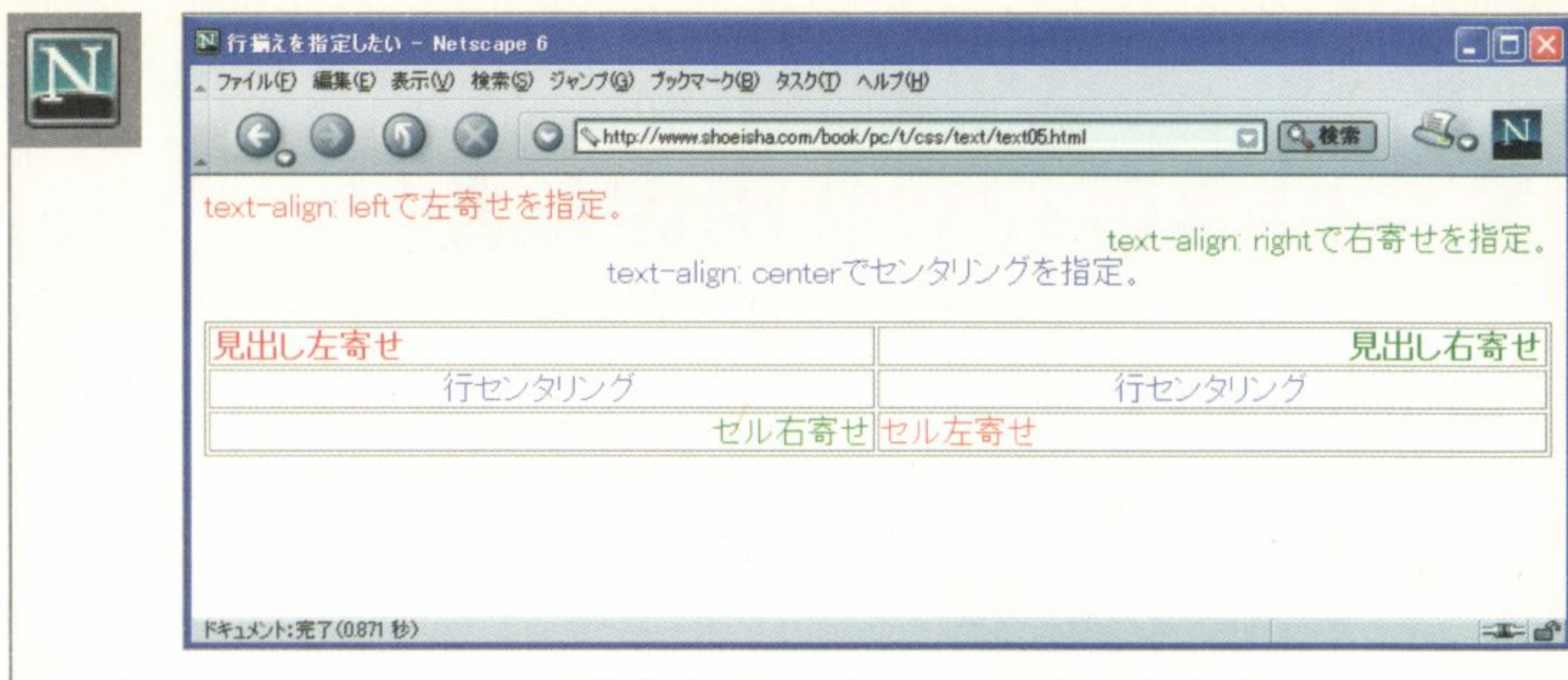
http://www.shoeisha.com/book/pc/t/css/text/text05.html

text-align: leftで左寄せを指定。

text-align: rightで右寄せを指定。

text-align: centerでセンタリングを指定。

見出し左寄せ	見出し右寄せ
行センタリング	行センタリング
セル右寄せ	セル左寄せ

ドキュメント：完了(0.871 秒)

行揃えを指定するHTMLタグをCSSに改める

HTMLタグで文字の行揃えを指定するには、次のようにalign属性を利用します。

<h△ align="★">〜</h△>	見出しの行揃え（△——見出しレベルを表す1〜6）
〜	段落の行揃え
〜	特定範囲の行揃え
<◇ align="★">〜</◇>	セル内の行揃え（◇——tr、th、td）
<center>〜</center>	任意の内容の行揃え

★——left、right、center

HTMLタグのalign属性の多くはDeprecated（推奨しない）とされており、行揃えはスタイルシートで指定することが推奨されています。またスタイルシートなら上記以外の要素の行揃えを指定することも可能です（ただしブロックレベル要素のみ）。同様の効果をスタイルシートで表現すると以下のようになります。

h△ { text-align: ★ }	見出しの行揃え（△——見出しレベルを表す1〜6）
p { text-align: ★ }	段落の行揃え
div { text-align: ★ }	特定範囲の行揃え
◇ { text-align: ★ }	セル内の行揃え（◇——tr、th、td）
☆ { text-align: center }	任意の内容の行揃え（☆——セレクタ）

★——left、right、center

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
left	○	○	○	○	○	○	○
center	○	○	○	○	○	○	○
right	○	○	○	○	○	○	○
justify	×	×*1	×*1	×*1	×	×	×

＊1：text-justifyと併用すれば○

※適用するセレクタによっても効果が変わります

参照
文字の垂直位置を指定したい……………………p.72
文字の均等割付を指定したい……………………p.76
要素の配置位置を指定したい……………………p.212
キャプションの位置を指定したい………………p.258

6 TEXT

文字の垂直位置を指定したい

vertical-align: ★

★……キーワード
サイズを表す数値＋単位
パーセントを表す数値＋%

インラインレベル要素とテーブルのセル要素（th 要素、td 要素）に設定し、それらの要素が表示される行の中での、垂直方向の位置（縦方向の位置）を指定します。

値には次のような指定方法があります。

キーワード

baseline	ベースラインに揃える（デフォルト）
super	上付き文字にする
sub	下付き文字にする
top	上に揃える
middle	中に揃える
bottom	下に揃える

サイズを表す数値＋単位

親要素のベースラインを0とし、そこからの垂直位置を単位付きの数値で設定します。正の値では上に、負の値では下に移動します。

パーセントを表す数値＋%

親要素のベースラインを0とし、そこからの垂直位置をline-height プロパティ（p.66 参照）で設定された高さに対する割合で指定します。正の値では上に、負の値では下に移動します。

テキスト

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>文字の垂直位置を指定したい</title>
<style type="text/css">
<!--
.sample1 { vertical-align: baseline }
.sample2 { vertical-align: sub }
.sample3 { vertical-align: super }
.sample5 { vertical-align: top }
.sample6 { vertical-align: middle }
.sample7 { vertical-align: bottom }
-->
</style>
</head>
<body>
<p>これは<span class="sample1">ベースライン（標準）</span>です。</p>
<p>これは<span class="sample2">下付き文字</span>です。</p>
<p>これは<span class="sample3">上付き文字</span>です。</p>
<p>topを指定<img src="cat2.gif" class="sample5"></p>
<p>middleを指定<img src="cat2.gif" class="sample6"></p>
<p>bottomを指定<img src="cat2.gif" class="sample7"></p>
</body>
</html>
```

▲Internet Explorer 6.0はインラインレベル要素に対するbottomは対応していません

垂直位置を指定するHTMLタグをCSSに改める

HTMLタグで要素の垂直位置を指定するには、次のような方法があります。

[～]	上付き文字
_～	下付き文字
<◇ valign="★">～</◇>	セル内の文字位置（◇――tr、th、td）
<img src="△" align="★">	画像と文字の関係（△――画像のURL）

★――top、middle、bottom、baseline

同様の効果をスタイルシートで表現すると以下のようになります。

☆	{ vertical-align: super }	上付き文字（☆――セレクタ）
☆	{ vertical-align: sub }	下付き文字（☆――セレクタ）
tr	{ vertical-align: ★ }	セル内の文字位置
th	{ vertical-align: ★ }	セル内の文字位置
td	{ vertical-align: ★ }	セル内の文字位置
img	{ vertical-align: ★}	画像と文字の関係

★――top、middle、bottom、baseline

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
baseline	○	○	○	○	△＊1	△＊1	○
sub	○	○	○	○	×	×	○
super	○	○	○	○	×	×	○
top	○	○	○	○	○	○	○
middle	○	○	○	○	○	○	○
bottom	○	○	△＊2	△＊2	○	○	○
サイズ	×	×	×	×	×	×	○
パーセント	×	×	×	×	×	×	○

＊1：セル要素に対しては×
＊2：インラインレベル要素に対しては×
※適用するセレクタによっても効果が変わります

行揃えを指定したい ……………………………p.69
要素の配置位置を指定したい…………………p.212

7 TEXT

文字の均等割付を指定したい

text-justify: ★

★……キーワード

文字の均等割付を指定します。

CSS2の仕様では、均等割付にはtext-align: justifyを指定するよう定義されていますが、この方法では均等割付にすることはできません。均等割付を適用するにはtext-align: justifyに加え、text-justifyプロパティを指定する必要があります。これはInternet Explorer5.5がW3Cの「Extensible Stylesheet Language (XSL)」の仕様を一部独自に採用したプロパティです。

値には以下のキーワードがあります。

auto	ブラウザに依存（デフォルト）
distribute	単語間隔と文字間隔の両方を調整して均等に割付ける
distribute-all-lines	単語間隔と文字間隔の両方を調整して均等に割付ける 最終行で行末まで満たないテキストも均等に割付ける
inter-cluster	単語間隔のないテキストを調整して均等に割付ける
inter-ideograph	漢字などの文字間隔と単語間隔の両方を調整して均等に割付ける
inter-word	単語間隔のみを調整して均等に割付ける。最終行で行末まで満たないテキストは均等割付にはならない
newspaper	単語間隔と文字間隔の両方を調整して均等に割付ける

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>文字の均等割付を指定したい</title>
<style type="text/css">
<!--
p           {
        font-family: Arial,Helvetica,sans-serif;
        font-size: 10pt;
        line-height: 140%;
        text-align: justify
}
#sample1    { text-justify: auto }
#sample2    { text-justify: distribute }
#sample3    { text-justify: distribute-all-lines }
#sample4    { text-justify: inter-cluster }
#sample5    { text-justify: inter-ideograph }
#sample6    { text-justify: inter-word }
#sample7    { text-justify: newspaper }
span        { font-weight: bold }
-->
</style>
</head>
<body>
<p id="sample1"><span>auto </span>Default. Allows the browser to determine which
justification algorithm to apply.<br>
ブラウザに依存します。</p>

<p id="sample2"><span>distribute </span>Handles spacing much like the newspaper
value. This form of justification is optimized for documents in Asian languages, such as
Thai.<br>
単語間隔と文字間隔の両方を調整して均等に割付けます。タイ語のようなアジア系の言語に
```

```
適した指定方法です。</p>

<p id="sample3"><span>distribute-all-lines </span>Justifies lines of text in the same way as the distribute value, except that it also justifies the last line of the paragraph. This form of justification is intended for ideographic text.<br>
単語間隔と文字間隔の両方を調整して均等に割付けます。最終行で行末まで満たないテキストも均等に割付ます。</p>

<p id="sample4"><span>inter-cluster </span>Justifies lines of text that contain no inter-word spacing. This form of justification is optimized for documents in Asian languages.<br>
単語間隔のないテキストを調整して均等に割付けます。アジア系の言語などに適した指定方法です。
</p>

<p id="sample5"><span>inter-ideograph </span>Justifies lines of ideographic text, and increases or decreases both inter-ideograph and inter-word spacing.<br>
漢字などの文字間隔と単語間隔の両方を調整して均等に割付けます。</p>

<p id="sample6"><span>inter-word </span>Aligns text by increasing the space between words. This value's spacing behavior is the fastest way to make all lines of text equal in length. Its justification behavior does not affect the last line of the paragraph.<br>
単語間隔のみを調整して均等に割付けます。最終行で行末まで満たないテキストは均等割付にはなりません。</p>

<p id="sample7"><span>newspaper </span>Increases or decreases the spacing between letters and between words. It is the most sophisticated form of justification for Latin alphabets.<br>
文字間隔と単語間隔の両方を調整して均等に割付けます。英文などに最適な指定方法です。
</p>

</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
共通	×	○	○	○	×	×	×

※適用する言語によっても対応は異なります
※ Macintosh版IE5は対応していません

auto Default. Allows the browser to determine which justification algorithm to apply.
ブラウザに依存します。

distribute Handles spacing much like the newspaper value. This form of justification is optimized for documents in Asian languages, such as Thai.
単語間隔と文字間隔の両方を調整して均等に割付けます。タイ語のようなアジア系の言語に適した指定方法です。

distribute-all-lines Justifies lines of text in the same way as the distribute value, except that it also justifies the last line of the paragraph. This form of justification is intended for ideographic text.
単語間隔と文字間隔の両方を調整して均等に割付けます。最終行で行末まで満たないテキストも均等に割付ます。

inter-cluster Justifies lines of text that contain no inter-word spacing. This form of justification is optimized for documents in Asian languages.
単語間隔のないテキストを調整して均等に割付けます。アジア系の言語などに適した指定方法です。

inter-ideograph Justifies lines of ideographic text, and increases or decreases both inter-ideograph and inter-word spacing.
漢字などの文字間隔と単語間隔の両方を調整して均等に割付けます。

inter-word Aligns text by increasing the space between words. This value's spacing behavior is the fastest way to make all lines of text equal in length. Its justification behavior does not affect the last line of the paragraph.
単語間隔のみを調整して均等に割付けます。最終行で行末まで満たないテキストは均等割付にはなりません。

newspaper Increases or decreases the spacing between letters and between words. It is the most sophisticated form of justification for Latin alphabets.
文字間隔と単語間隔の両方を調整して均等に割付けます。英文などに最適な指定方法です。

ページが表　インターネット

auto Default. Allows the browser to determine which justification algorithm to apply.
ブラウザに依存します。

distribute Handles spacing much like the newspaper value. This form of justification is optimized for documents in Asian languages, such as Thai.
単語間隔と文字間隔の両方を調整して均等に割付けます。タイ語のようなアジア系の言語に適した指定方法です。

distribute-all-lines Justifies lines of text in the same way as the distribute value, except that it also justifies the last line of the paragraph. This form of justification is intended for ideographic text.
単語間隔と文字間隔の両方を調整して均等に割付けます。最終行で行末まで満たないテキストも均等に割付ます。

inter-cluster Justifies lines of text that contain no inter-word spacing. This form of justification is optimized for documents in Asian languages.
単語間隔のないテキストを調整して均等に割付けます。アジア系の言語などに適した指定方法です。

inter-ideograph Justifies lines of ideographic text, and increases or decreases both inter-ideograph and inter-word spacing.
漢字などの文字間隔と単語間隔の両方を調整して均等に割付けます。

inter-word Aligns text by increasing the space between words. This value's spacing behavior is the fastest way to make all lines of text equal in length. Its justification behavior does not affect the last line of the paragraph.
単語間隔のみを調整して均等に割付けます。最終行で行末まで満たないテキストは均等割付にはなりません。

newspaper Increases or decreases the spacing between letters and between words. It is the most sophisticated form of justification for Latin alphabets.
文字間隔と単語間隔の両方を調整して均等に割付けます。英文などに最適な指定方法です。

ドキュメント:完了(0...

▲Netscapeは対応していません

行揃えを指定したい ……………………………p.69
文字間隔を指定したい …………………………p.80

8 TEXT

文字間隔を指定したい

letter-spacing: ★

★……サイズを表す数値＋単位
キーワード

文字と文字の間隔（字間）を指定します。
値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけて文字間隔を指定します。単位についてはp.46を参照してください。

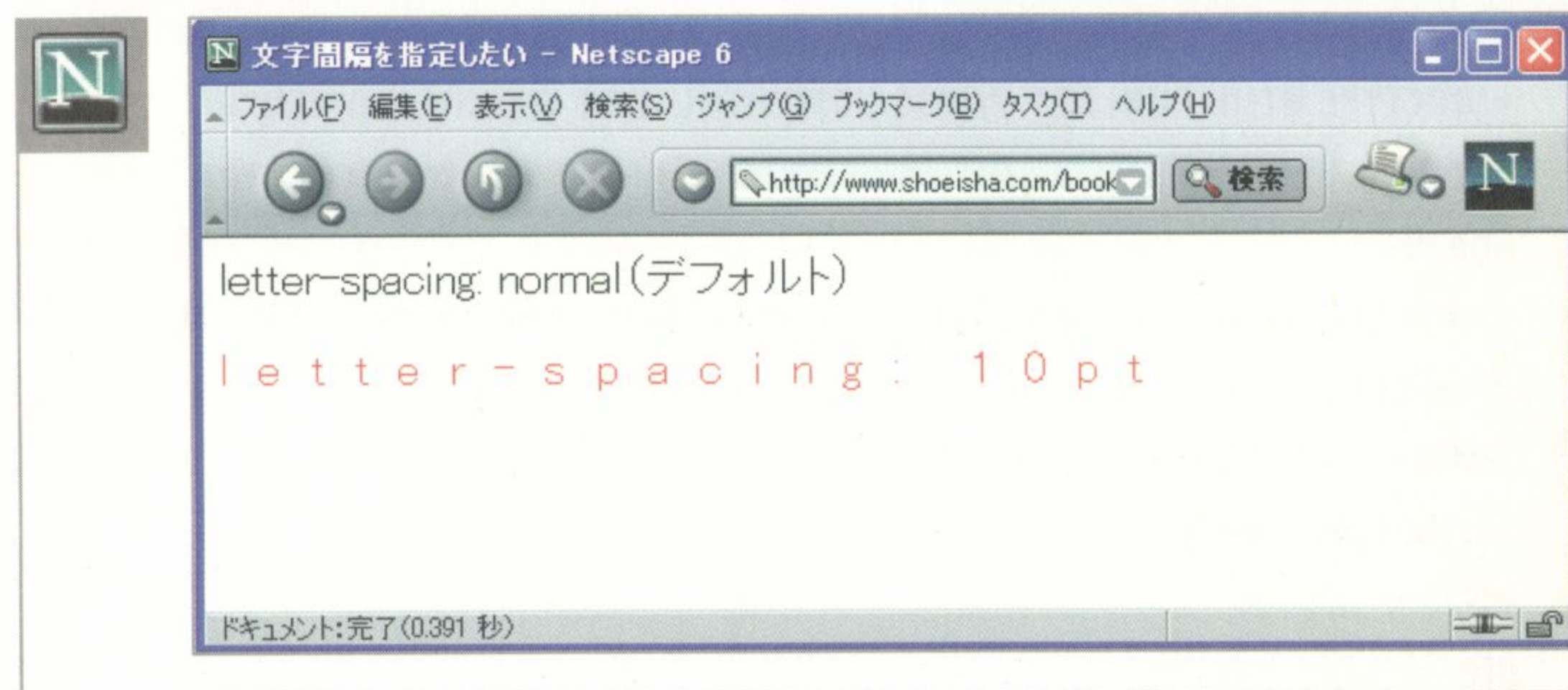
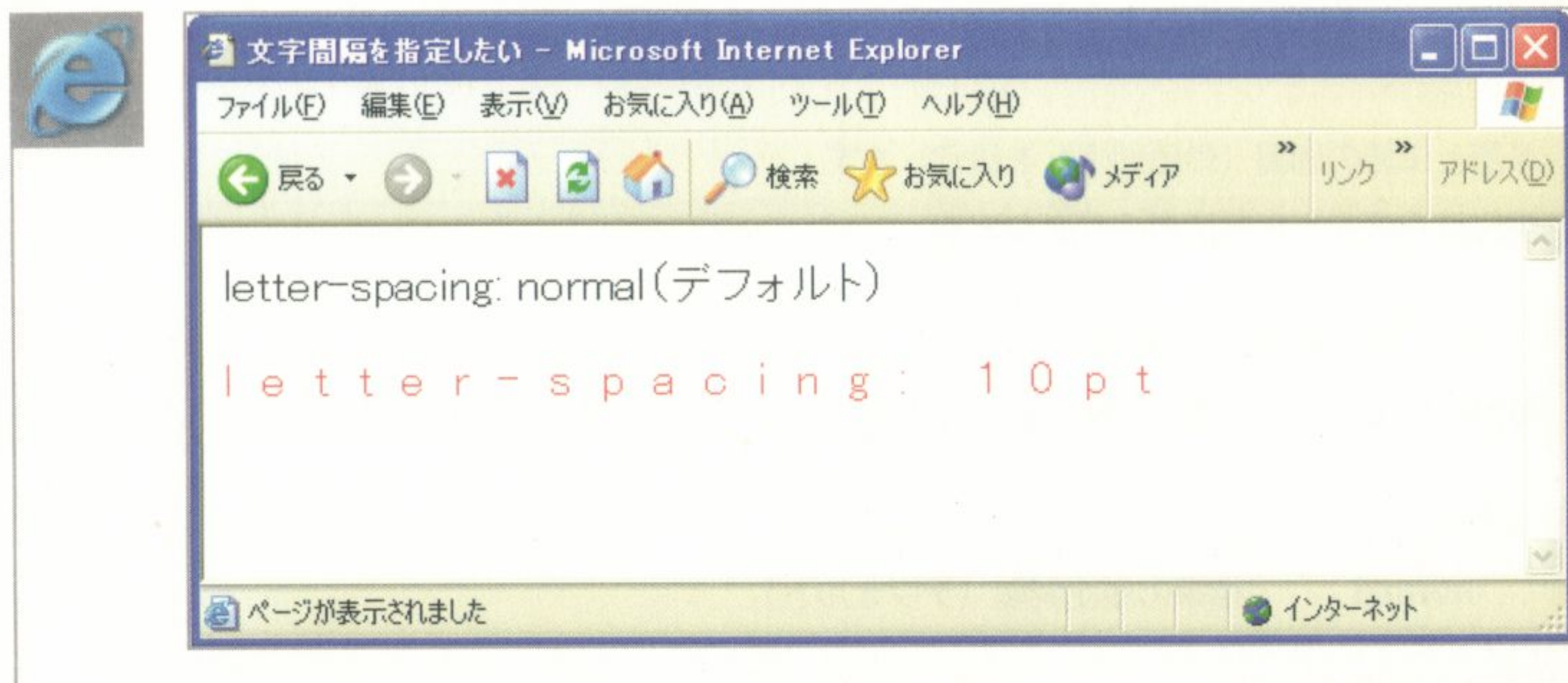
マイナスの値を指定すると文字どうしが重なって表示されます。この性質を利用して文字列を重ね合わせたデザインを表現することもできます。

キーワード

normal	標準の文字間隔（デフォルト）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>文字間隔を指定したい</title>
<style type="text/css">
<!--
p.sample1     { letter-spacing: normal }
p.sample2     {
        letter-spacing: 10pt;
        color: red
}
-->
```

```
</style>
</head>
<body>
<p class="sample1">letter-spacing: normal（デフォルト）</p>
<p class="sample2">letter-spacing: 10pt</p>
</body>
</html>
```


	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	×	×	○
normal	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

単語間隔を指定したい ……………………………p.82
文字をグリッドにおさめたい…………………p.108

9 TEXT

単語間隔を指定したい

word-spacing: ★

★……サイズを表す数値＋単位
キーワード

単語と単語の間隔（単語間隔）を設定します。
値には次のような指定方法があります。

サイズを表す数値＋単位

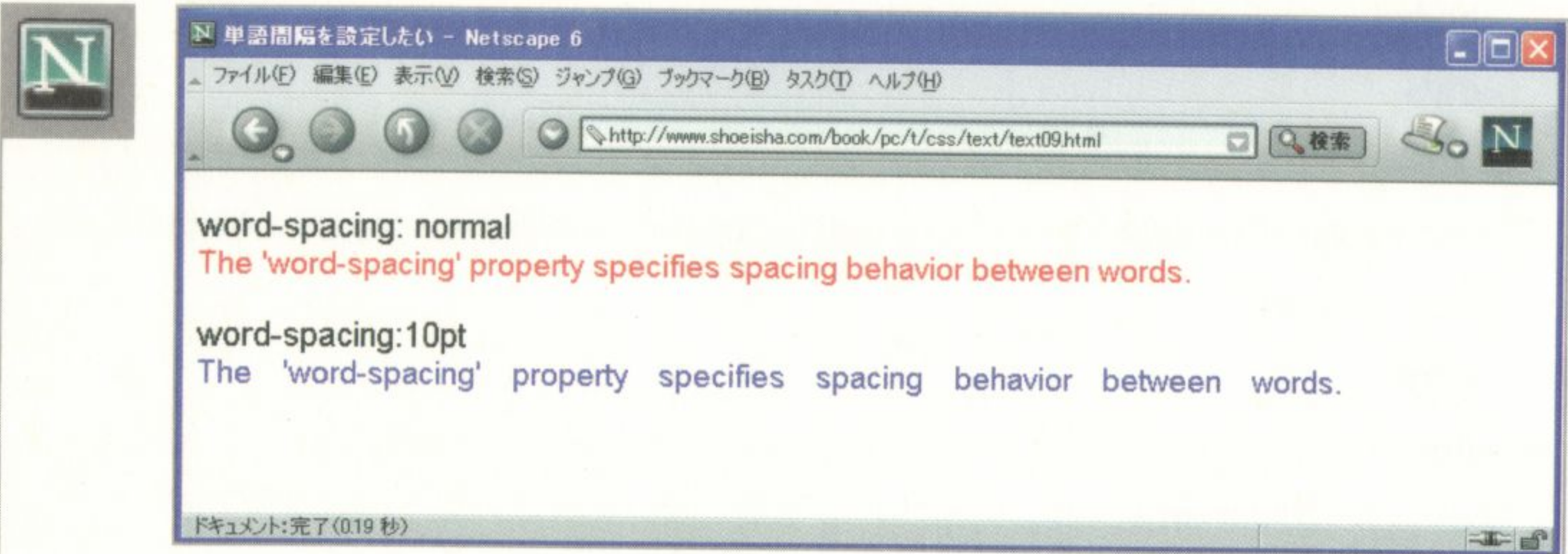
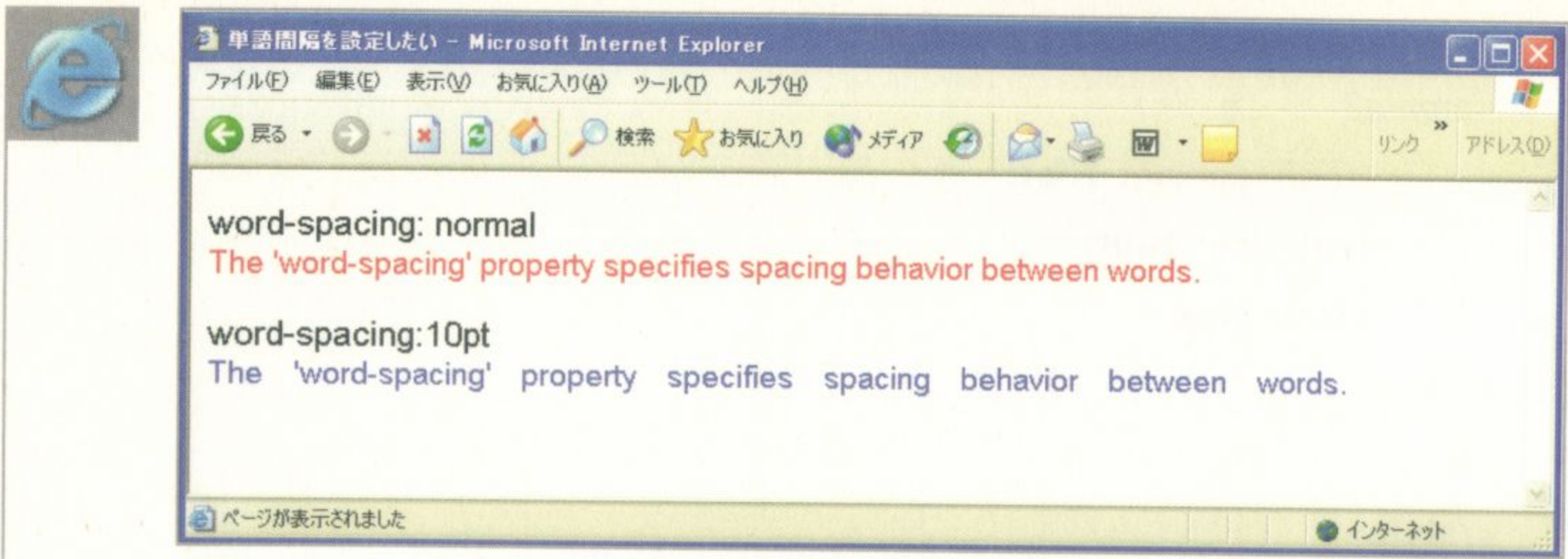
数値に単位をつけて単語間隔を設定します。単位についてはp.46を参照してください。

キーワード

normal　標準の単語間隔（デフォルト）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>単語間隔を設定したい</title>
<style type="text/css">
<!--
div        {
      margin: 20px auto;
      font-family: Arial,Helvetica,sans-serif
}
.sample1    {
      word-spacing: normal;
      font-size: 14pt;
      color: red
```

```
}
.sample2        {
        word-spacing: 10pt;
        font-size: 14pt;
        color: blue
}
-->
</style>
</head>
<body>
<div>
word-spacing: normal<br>
<span class="sample1">The 'word-spacing' property specifies spacing behavior between
words. </span>
</div>
<div>
word-spacing:10pt<br>
<span class="sample2">The 'word-spacing' property specifies spacing behavior between
words. </span>
</div>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	×	×	×	○	×	×	○
normal	×	×	×	○	×	×	○

※適用するセレクタによっても効果が変わります

※ Macintosh 版 IE5 は対応しています

参照 文字間隔を設定したい ……………………… p.80

1行目にインデントを設定したい

text-indent: ★

★••••••サイズを表す数値＋単位
パーセントを表す数値＋%

ブロックレベル要素の中にある文章の1行目のインデント（字下げ）を設定します。
値には次のような指定方法があります。

サイズを表す数値＋単位

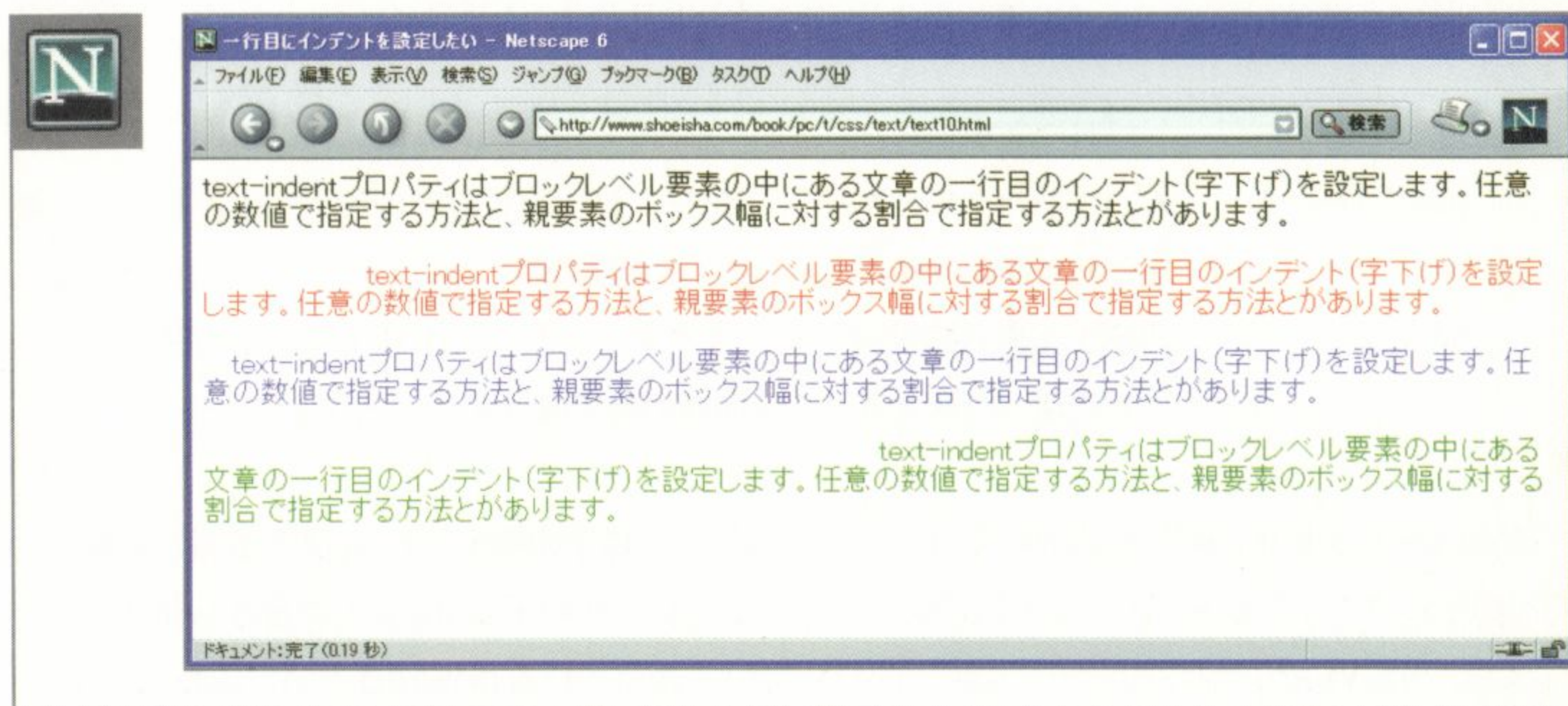
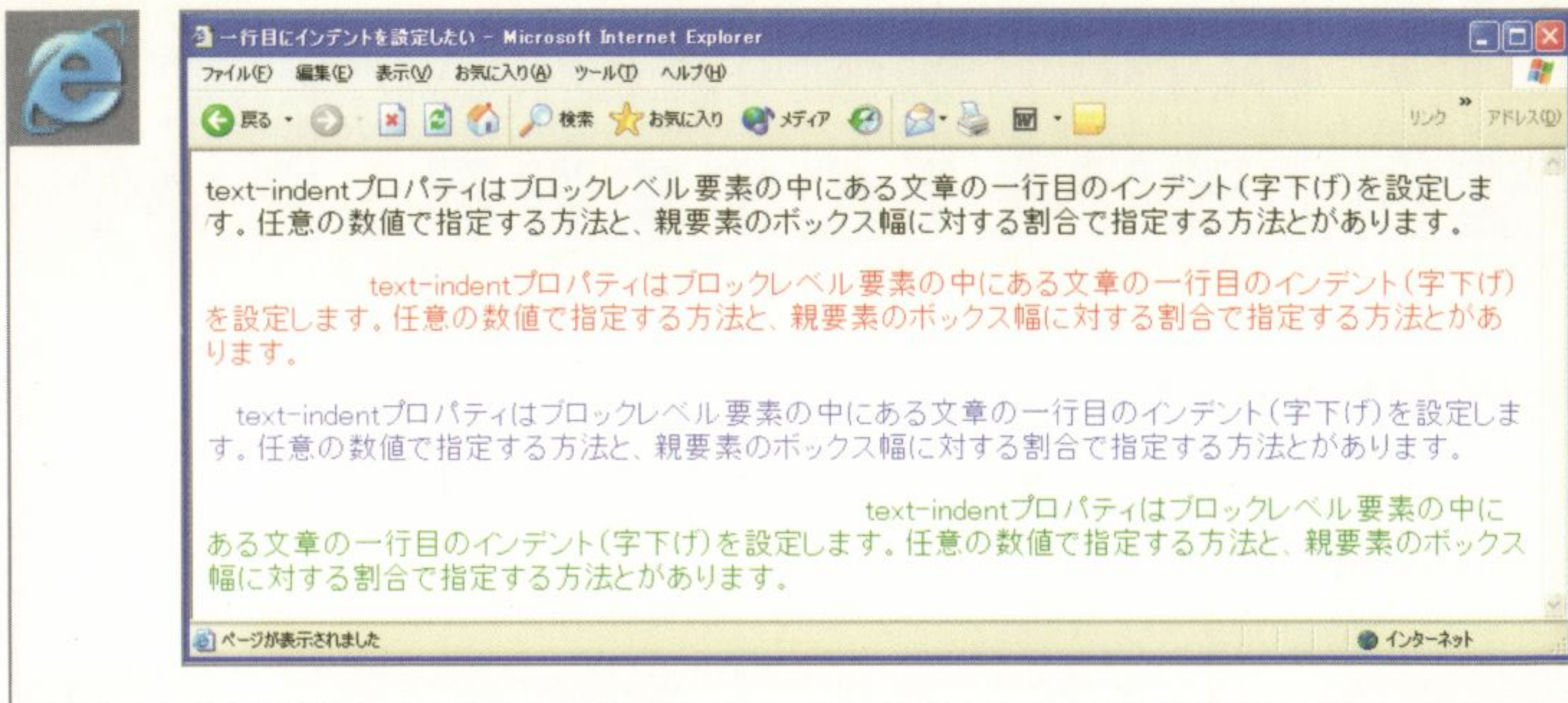
数値に単位をつけてインデントの幅を指定します。単位についてはp.46を参照してください。

パーセントを表す数値＋%

親要素のボックス領域の幅に対する割合で設定します。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>一行目にインデントを設定したい</title>
<style type="text/css">
<!--
p.sample1     {
        text-indent: 3cm;
        color: red
}
p.sample2     {
        text-indent: 1em;
        color: blue
}
```

```
p.sample3      {
        text-indent: 50%;
        color: green
}
-->
</style>
</head>

<body>

<p>text-indent プロパティはブロックレベル要素の中にある文章の一行目のインデント（字下げ）を設定します。任意の数値で指定する方法と、親要素のボックス幅に対する割合で指定する方法とがあります。</p>

<p class="sample1">text-indent プロパティはブロックレベル要素の中にある文章の一行目のインデント（字下げ）を設定します。任意の数値で指定する方法と、親要素のボックス幅に対する割合で指定する方法とがあります。</p>

<p class="sample2">text-indent プロパティはブロックレベル要素の中にある文章の一行目のインデント（字下げ）を設定します。任意の数値で指定する方法と、親要素のボックス幅に対する割合で指定する方法とがあります。</p>

<p class="sample3">text-indent プロパティはブロックレベル要素の中にある文章の一行目のインデント（字下げ）を設定します。任意の数値で指定する方法と、親要素のボックス幅に対する割合で指定する方法とがあります。</p>

</body>

</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

マージンを個別に指定したい……………………p.160
パディングを個別に指定したい…………………p.167

11 TEXT

空白や改行をそのまま表示したい

white-space: ★

★……キーワード

要素内の半角スペースやタブ、改行をそのまま表示させます。内容も入力したとおりに表示されます。

値には以下のキーワードがあります。

pre	入力された半角スペースやタブ、改行をそのまま表示。内容も入力されたとおりに表示する。折り返しも行わない
normal	連続する複数の半角スペースやタブ、改行文字を1つの半角スペースとして処理。内容はボックス領域の端で改行する（デフォルト）

Windows版Internet Explorerでは、バージョン6の標準準拠モード（p.51参照）の場合にのみ動作します。互換モードやそれ以前のバージョンのInternet Explorerでは動作しません。

なお、white-spaceプロパティには、改行しないで表示させるnowrapという値もあります。nowrapについては次項を参照してください。

SOURCE

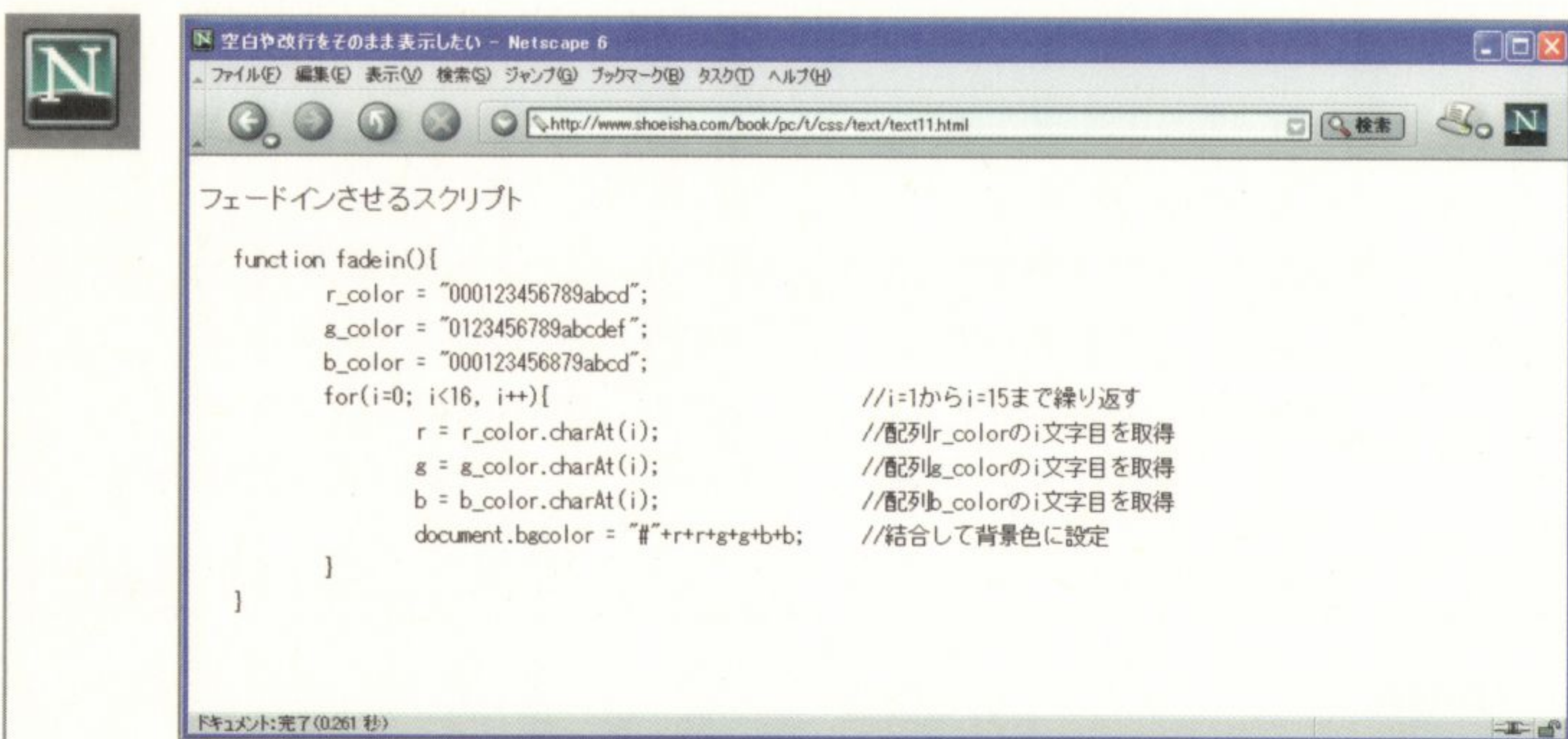
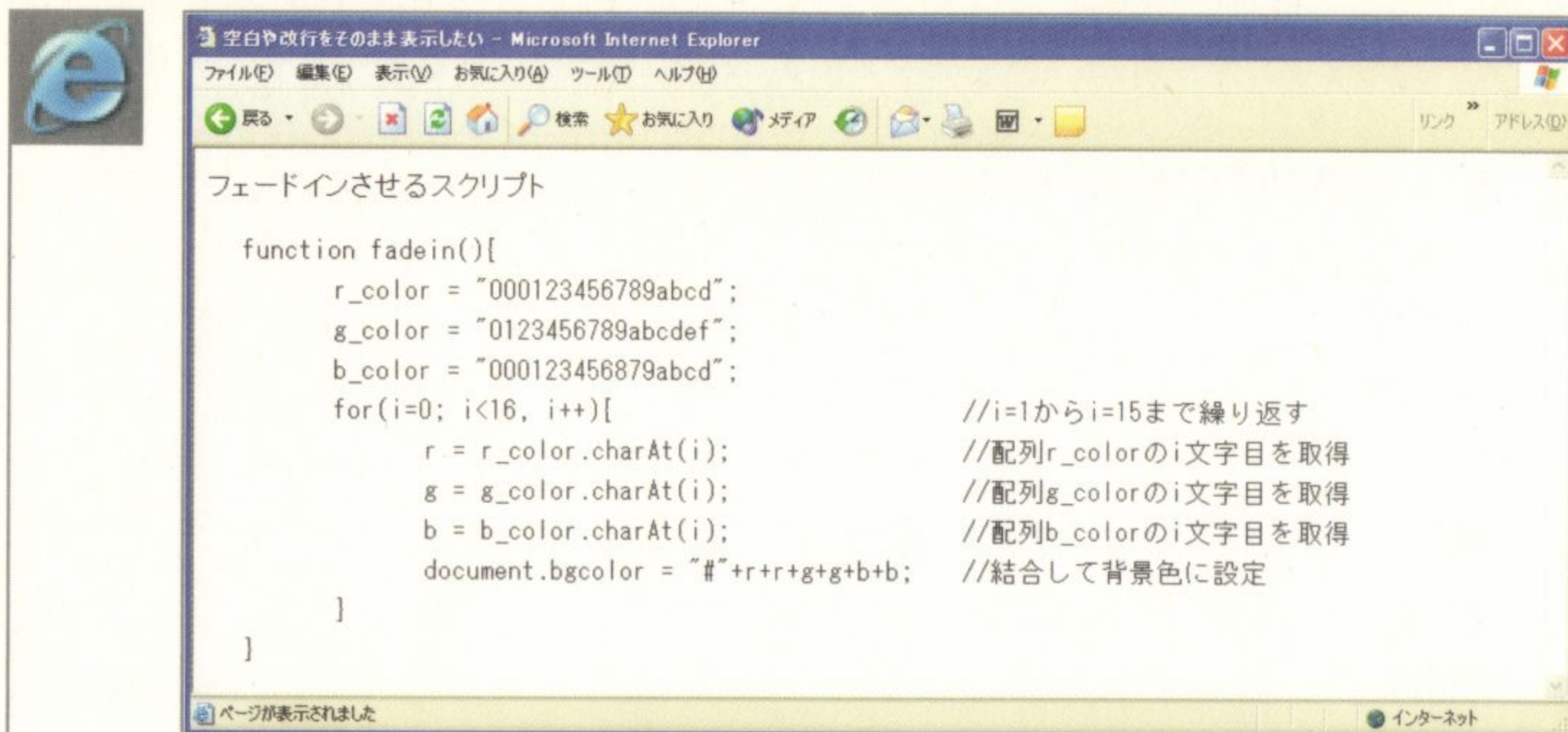
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>空白や改行をそのまま表示したい</title>
<style type="text/css">
<!--
code  {
        white-space: pre;
```

```
        line-height: 150%
}
div     { margin: auto 25px }
-->
</style>
</head>
<body>
<p>フェードインさせるスクリプト</p>
<div>
<code>
function fadein()   {
        r_color = "000123456789abcd";
        g_color = "0123456789abcdef";
        b_color = "000123456879abcd";
                for(i=0; i<16, i++){                //i=1からi=15まで繰り返す
                r = r_color.charAt(i);              //配列r_colorのi文字目を取得
                g = g_color.charAt(i);              //配列g_colorのi文字目を取得
                b = b_color.charAt(i);              //配列b_colorのi文字目を取得
                document.bgcolor = "#"+r+r+g+g+b+b;         //結合して背景色に設定
        }
}
</code>
</div>
</body>
</html>
```

フェードインさせるスクリプト

```
function fadein(){
        r_color = "000123456789abcd";
        g_color = "0123456789abcdef";
        b_color = "000123456879abcd";
        for(i=0; i<16, i++){                    //i=1からi=15まで繰り返す
                r = r_color.charAt(i);          //配列r_colorのi文字目を取得
                g = g_color.charAt(i);          //配列g_colorのi文字目を取得
                b = b_color.charAt(i);          //配列b_colorのi文字目を取得
                document.bgcolor = "#"+r+r+g+g+b+b;  //結合して背景色に設定
        }
}
```

フェードインさせるスクリプト

```
function fadein(){
        r_color = "000123456789abcd";
        g_color = "0123456789abcdef";
        b_color = "000123456879abcd";
        for(i=0; i<16, i++){                    //i=1からi=15まで繰り返す
                r = r_color.charAt(i);          //配列r_colorのi文字目を取得
                g = g_color.charAt(i);          //配列g_colorのi文字目を取得
                b = b_color.charAt(i);          //配列b_colorのi文字目を取得
                document.bgcolor = "#"+r+r+g+g+b+b;  //結合して背景色に設定
        }
}
```

空白や改行をそのまま表示するHTMLタグをCSSに改める

HTMLタグで空白や改行をそのまま表示する指定をするには、次の方法があります。

<pre>～</pre>　　入力したまま表示

同様の効果をスタイルシートで表現すると以下のようになります。

★ { white-space: pre }　　入力したまま表示（★――セレクタ）

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
normal	×	×	○	○	○	○	○
pre	×	×	×	○*	○	○	○

＊標準準拠モードの場合に動作します。
※適用するセレクタによっても効果が変わります
※Macintosh版IE5はpreに対応しています

禁則処理を適用したい ……………………p.94
単語内での改行処理を指定したい ……………p.97

改行しないで表示させたい

white-space: ★

★••••••キーワード

要素内の半角スペースやタブ、改行をまとめ、内容を改行しないで表示させます。

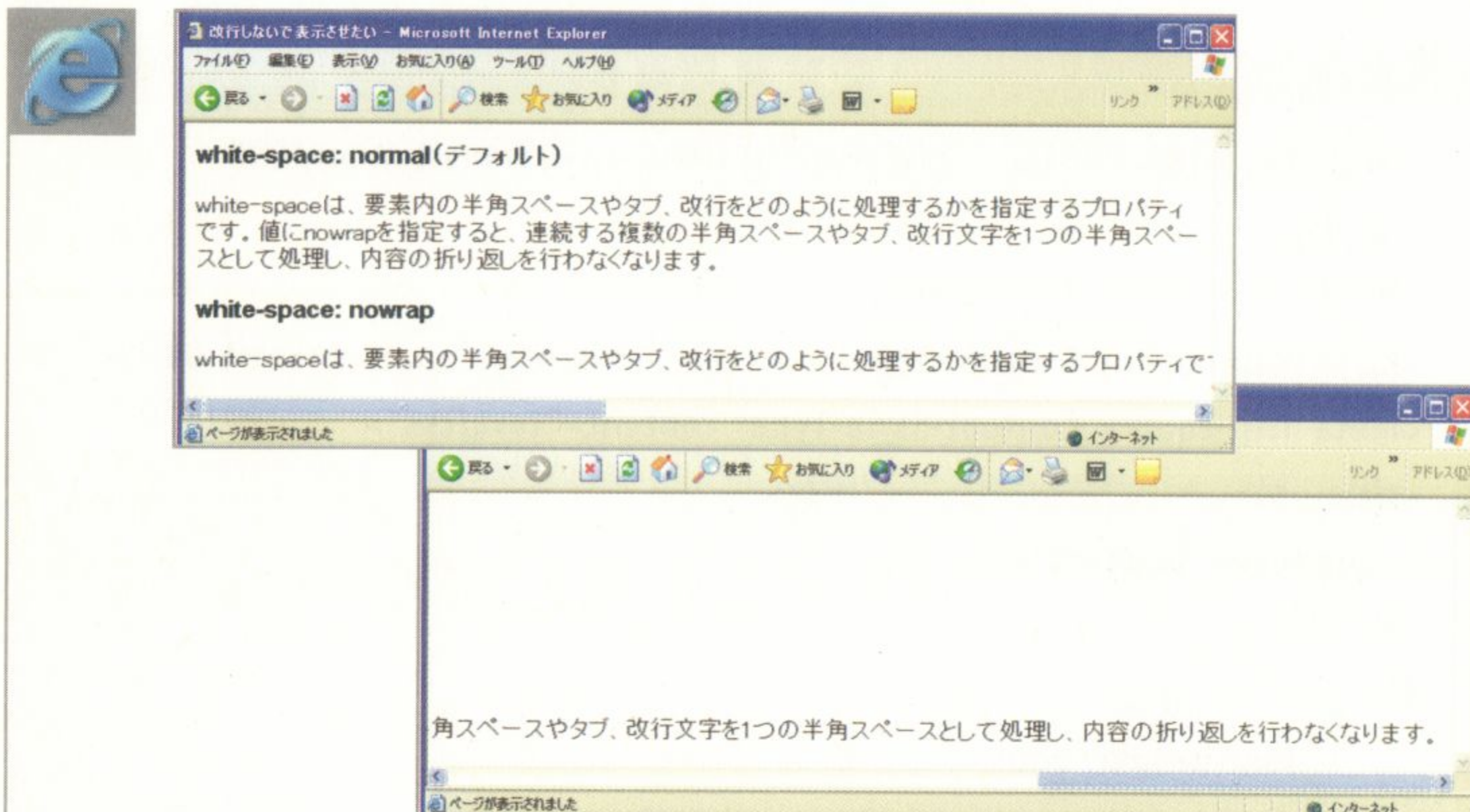
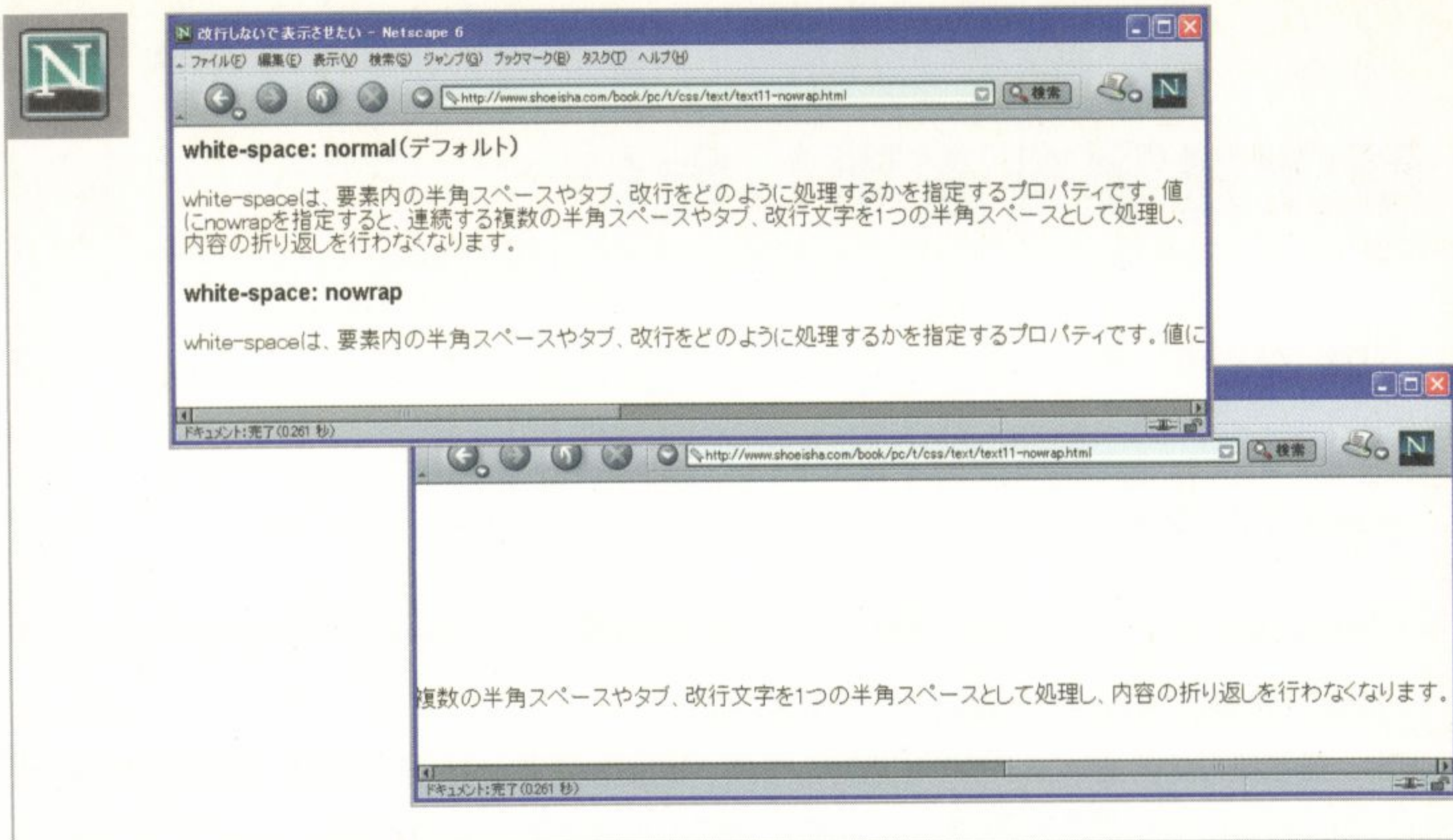
値には以下のキーワードがあります。

nowrap	連続する複数の半角スペースやタブ、改行文字を１つの半角スペースとして処理。内容の折り返しを行わない
normal	連続する複数の半角スペースやタブ、改行文字を１つの半角スペースとして処理。内容はボックス領域の端で改行する（デフォルト）

なお、white-space プロパティには、空白や改行をそのまま表示させる pre という値もあります。pre については前項を参照してください。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>改行しないで表示させたい</title>
<style type="text/css">
<!--
div		{
	font-weight: bold;
	font-family: Arial,Helvetica,sans-serif
}
p#sample1	{ white-space: normal }
p#sample2	{ white-space: nowrap }
-->
```

```
</style>
</head>
<body>
<div>white-space: normal（デフォルト）</div>
<p id="sample1">
white-spaceは、要素内の半角スペースやタブ、改行をどのように処理するかを指定するプロパティです。値にnowrapを指定すると、連続する複数の半角スペースやタブ、改行文字を1つの半角スペースとして処理し、内容の折り返しを行わなくなります。
</p>
<div>white-space: nowrap</div>
<p id="sample2">
white-spaceは、要素内の半角スペースやタブ、改行をどのように処理するかを指定するプロパティです。値にnowrapを指定すると、連続する複数の半角スペースやタブ、改行文字を1つの半角スペースとして処理し、内容の折り返しを行わなくなります。
</p>
</body>
</html>
```

空白や改行の扱いを指定するHTMLタグをCSSに改める

HTMLタグで空白や改行の不可を指定するには、次の方法があります。

<nobr>～</nobr>	改行不可
<◇ nowrap>～</◇>	セル内の改行不可（◇──th、td）

<th>、<td>タグのnowrap属性はDeprecated（推奨しない）とされており、セル内の改行不可の指定はスタイルシートで指定することが推奨されています。また、<nobr>タグはHTML4.01規格外です。同様の効果をスタイルシートで表現すると以下のようになります。

★ { white-space: nowrap }	改行不可（★──セレクタ）
◇ { white-space: nowrap }	セル内の改行不可（◇──th、td）

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
normal	×	×	○	○	○	○	○
nowrap	×	×	○	○	×	×	○

※適用するセレクタによっても効果が変わります

※Macintosh版IE5は対応しています

空白や改行をそのまま表示したい ･････････････････p.88

13 TEXT

禁則処理を適用したい

line-break: ★

★……キーワード

日本語の禁則処理を適用するかどうかを指定します。

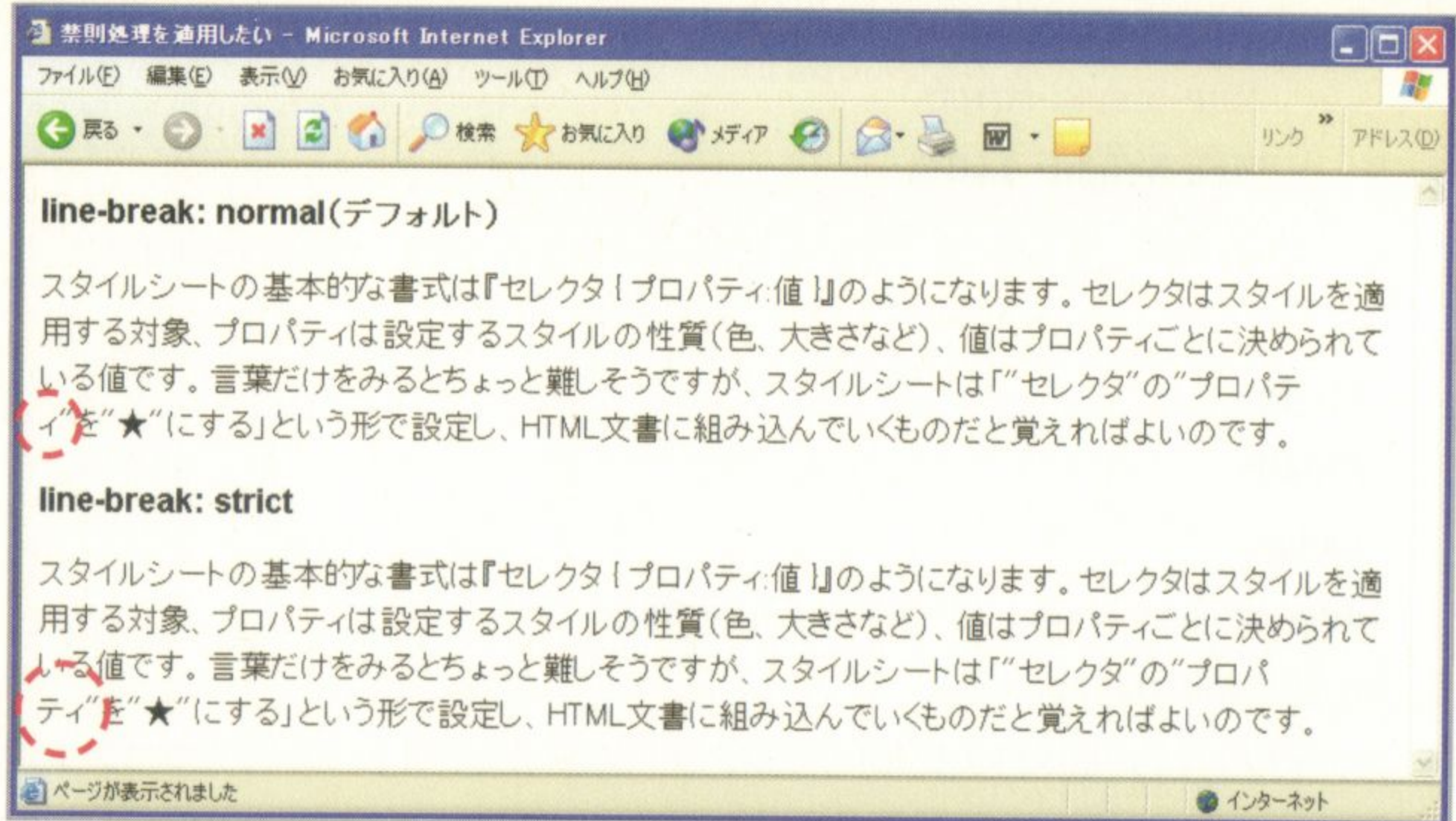
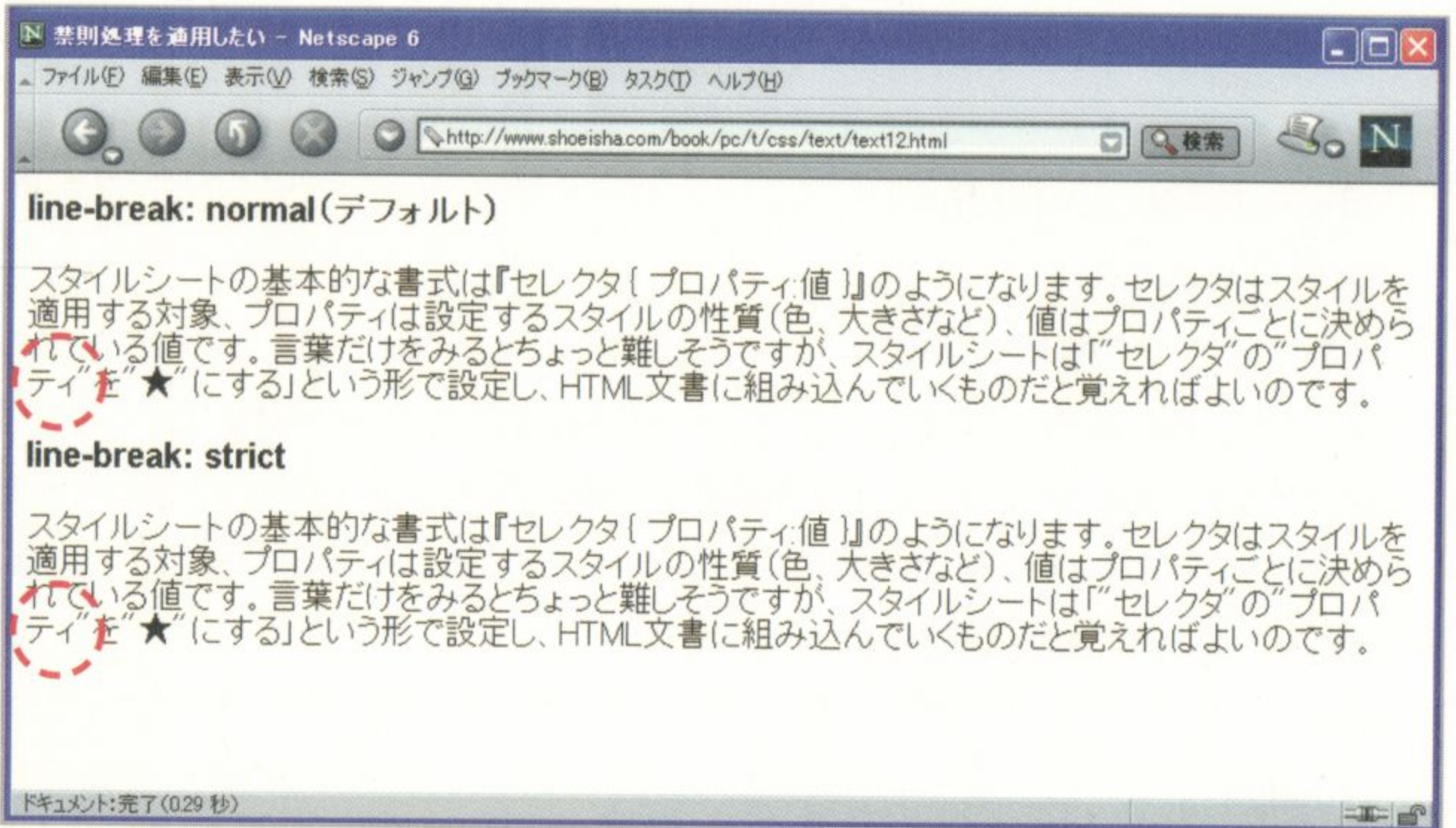
通常、禁則処理の方法はブラウザに依存します。Internet Explorerの場合、禁則処理が厳密ではなく、行頭に括弧の受けや句読点、促音、拗音の半音文字、音引記号などがくることがありますが、line-breakプロパティを指定すると厳密な禁則処理を行えるようになります。

値には以下のキーワードがあります。

normal	ブラウザ依存の禁則処理（デフォルト）
strict	厳密に禁則処理をする

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>禁則処理を適用したい</title>
<style type="text/css">
<!--
div          {
        font-weight: bold;
        font-family: Arial,Helvetica,sans-serif
}
p#sample1    {
        line-break: normal;
        line-height: 140%
}
```

```
p#sample2    {
        line-break: strict;
        line-height: 140%
}
-->
</style>
</head>
<body>
<div>line-break: normal（デフォルト）</div>
<p id="sample1">
スタイルシートの基本的な書式は『セレクタ { プロパティ: 値 }』のようになります。セレク
タはスタイルを適用する対象、プロパティは設定するスタイルの性質（色、大きさなど）、
値はプロパティごとに決められている値です。言葉だけをみるとちょっと難しそうですが、
スタイルシートは「"セレクタ"の"プロパティ"を"★"にする」という形で設定し、HTML文書
に組み込んでいくものだと覚えればよいのです。
</p>
<div>line-break: strict</div>
<p id="sample2">
スタイルシートの基本的な書式は『セレクタ { プロパティ: 値 }』のようになります。セレク
タはスタイルを適用する対象、プロパティは設定するスタイルの性質（色、大きさなど）、
値はプロパティごとに決められている値です。言葉だけをみるとちょっと難しそうですが、
スタイルシートは「"セレクタ"の"プロパティ"を"★"にする」という形で設定し、HTML文書
に組み込んでいくものだと覚えればよいのです。
</p>
</body>
</html>
```

▲Interner Explorer ではブラウザの禁則処理があいまいなので、「つ」や「よ」の小さな文字や長音符が行頭にくることがありますが、strict を適用すると禁則処理が厳密になります

▲Netscape ではブラウザが自動的に厳密な禁則処理を行いますので line-break プロパティによる変化はありません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
normal	×	○	○	○	×	×	×
strict	×	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります

※ Macintosh 版 IE5 はブラウザが自動的に厳密な禁則処理を行います

参照

単語内での改行処理を指定したい ··················p.97

14 TEXT

単語内での改行処理を指定したい

word-break: ★

★……キーワード

単語内における改行の方法を指定します。

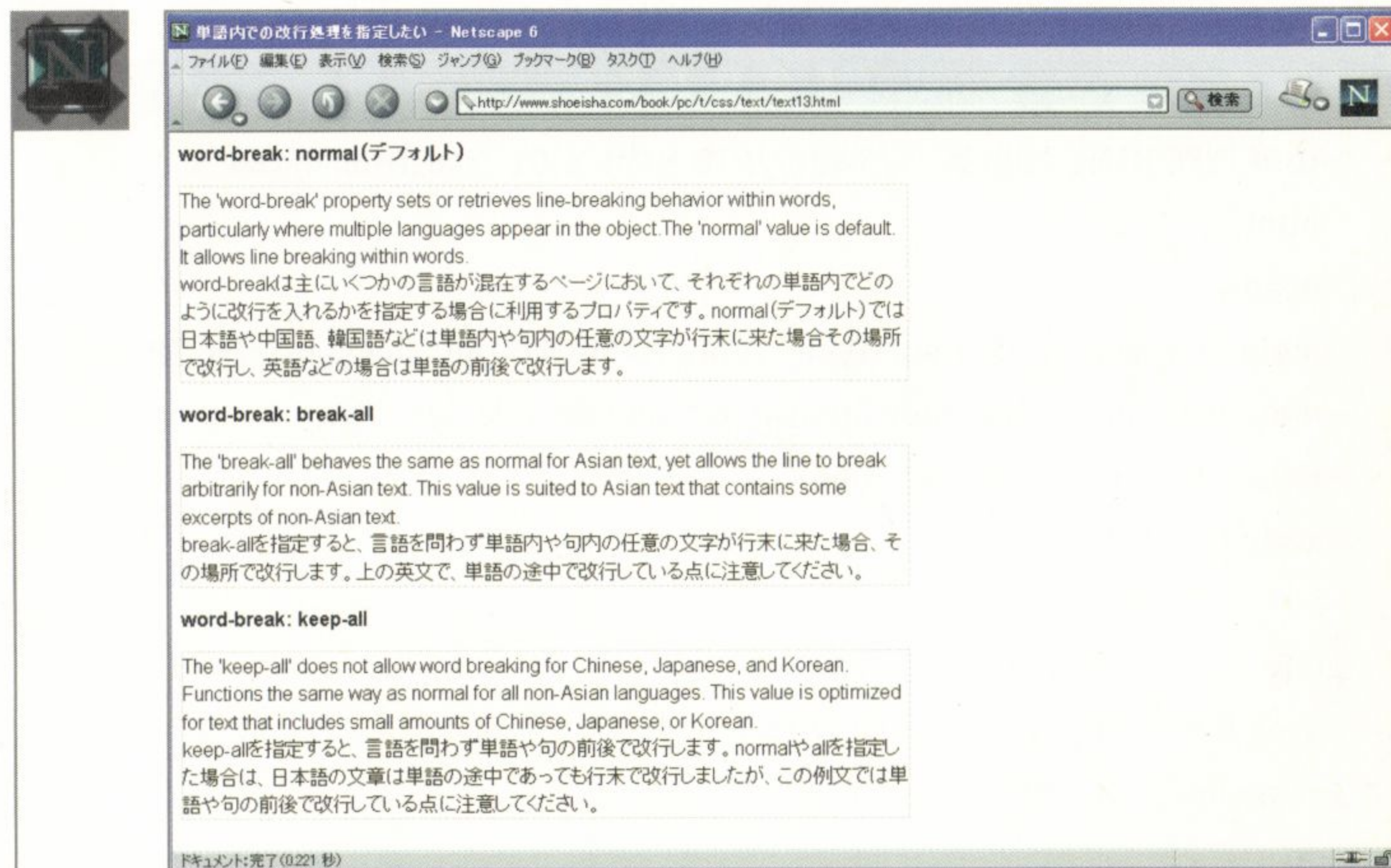
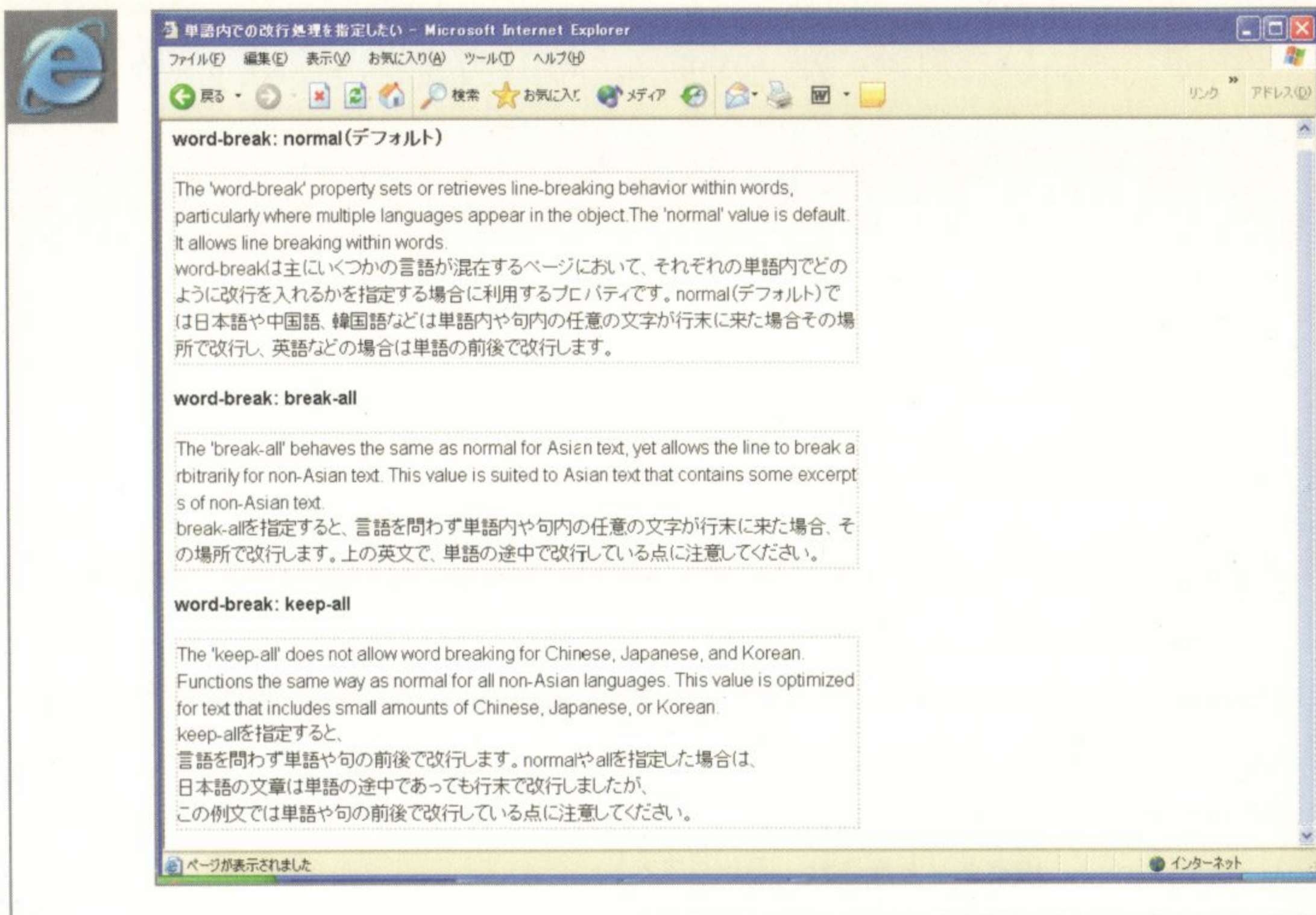
主にいくつかの言語が混在するページで、それぞれの単語内でどのように改行を入れるかを指定する場合に利用します。

値には以下のキーワードがあります。

normal	日本語や中国語、韓国語などは単語内や句内の任意の文字が行末に来た場合その場所で改行する。英語などの場合は単語の前後で改行する（デフォルト）
break-all	言語を問わず単語内や句内の任意の文字が行末に来た場合、その場所で改行する
keep-all	言語を問わず単語や句の前後で改行する

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>単語内での改行処理を指定したい</title>
<style type="text/css">
<!--
body      { font-family: Arial,Helvetica,sans-serif }
div       {
        font-size: medium;
        width: 600px
}
p         {
```

```
        width: 600px;
        border: thin dotted #c0c0c0;
        line-height: 140%
 }
#sample1 { word-break: normal }
#sample2 { word-break: break-all }
#sample3 { word-break: keep-all }
-->
</style>
</head>
<body>

<div>word-break: normal（デフォルト）</div>
<p id="sample1">
The 'word-break' property sets or retrieves line-breaking behavior within ……（中略）……単
語の前後で改行します。
</p>

<div>word-break: break-all</div>
<p id="sample2">
The 'break-all' behaves the same as normal for Asian text, yet allows the ……（中略）……上
の英文で、単語の途中で改行している点に注意してください。
</p>

<div>word-break: keep-all</div>
<p id="sample3">
The 'keep-all' does not allow word breaking for Chinese, Japanese, and ……（中略）……こ
の例文では単語や句の前後で改行している点に注意してください。
</p>

</body>

</html>
```

▲Netscapeは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
normal	×	○	○	○	×	×	×
break-all	×	○	○	○	×	×	×
keep-all	×	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります
※Macintosh版IE5は対応していません

参照 禁則処理を適用したい ····························p.94

15 TEXT

縦書きで表示したい

writing-mode: ★

★……キーワード

ページを縦書きで表示するよう指定します。

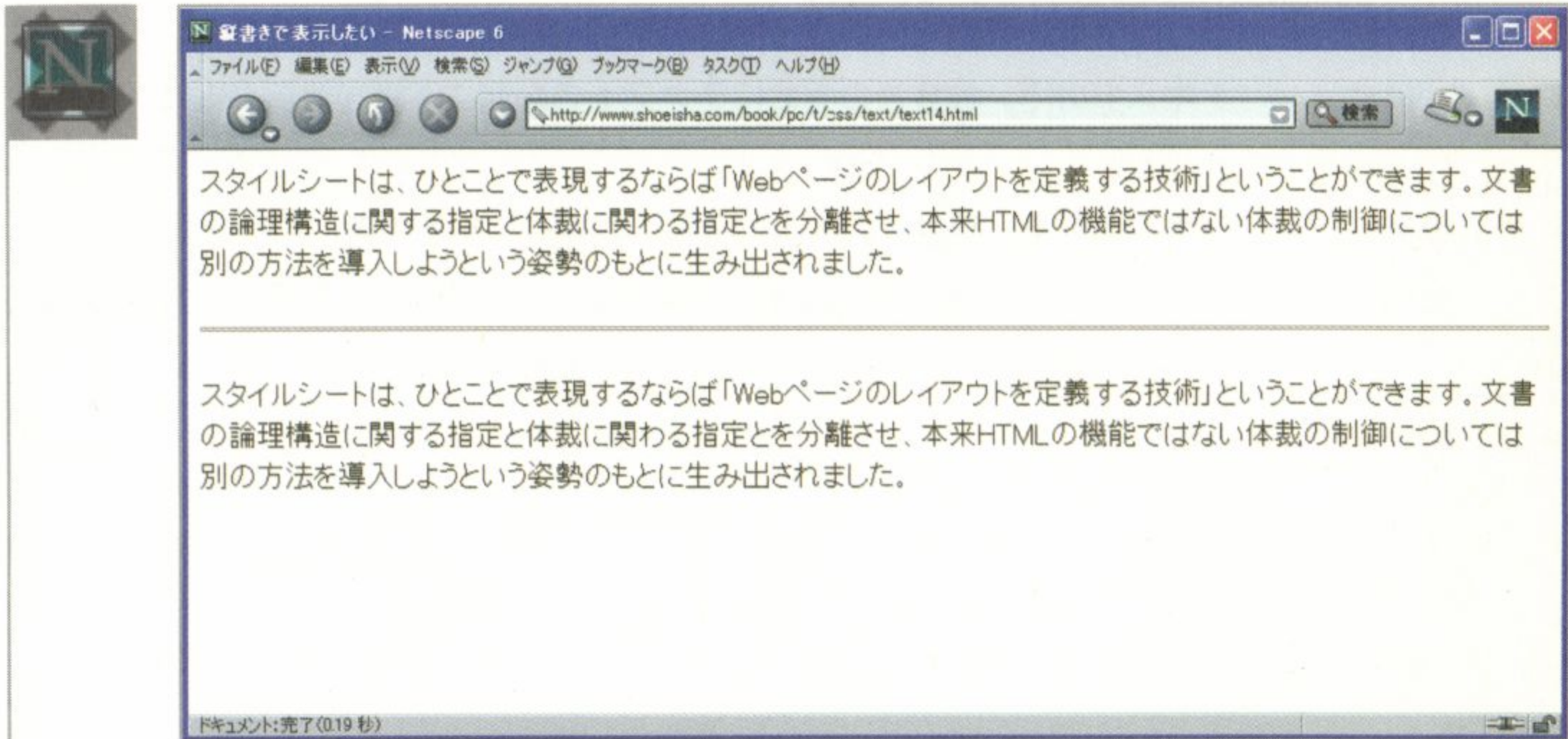
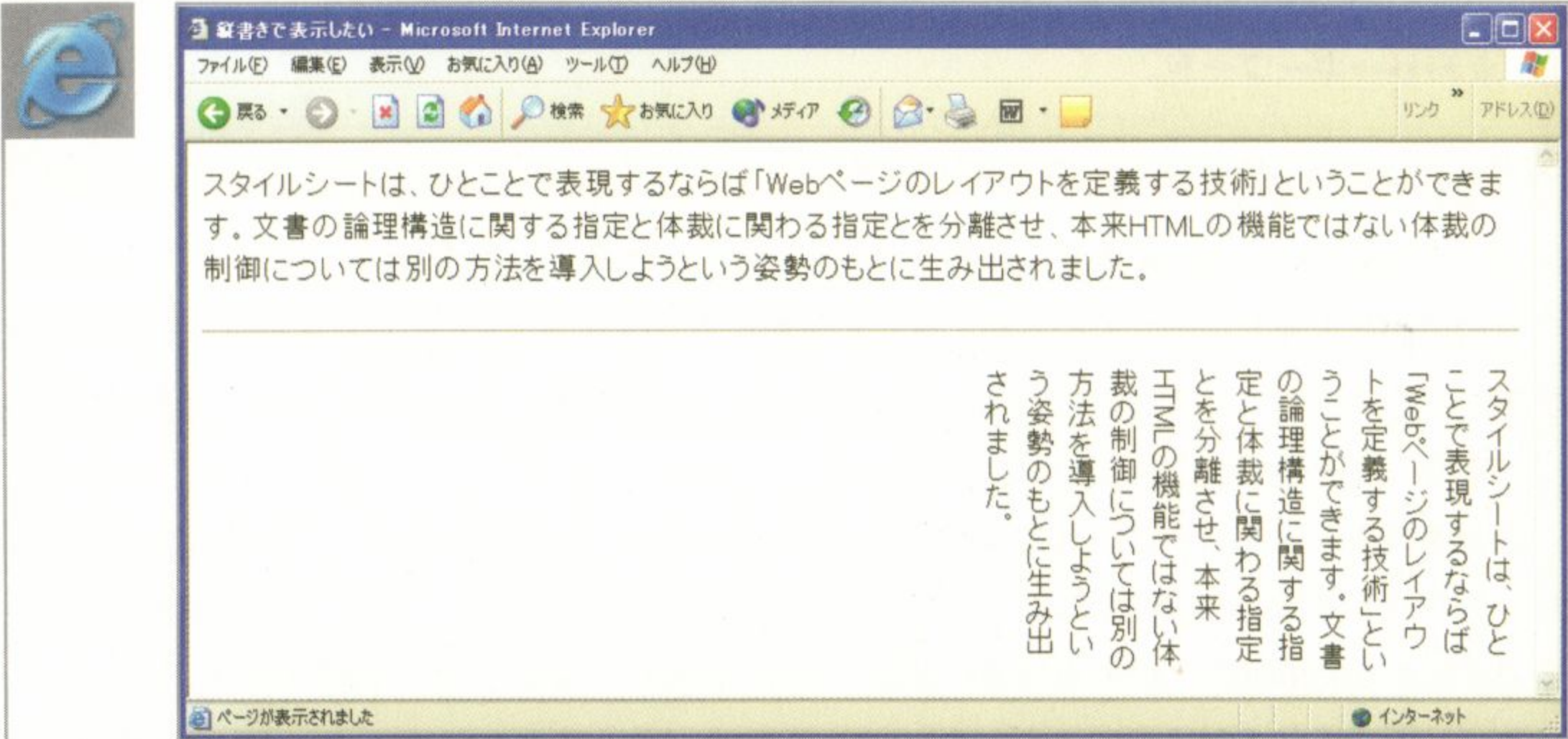
Internet Explorer5.5がW3Cの「Extensible Stylesheet Language（XSL）」の仕様を一部独自に採用したプロパティです。

値には以下のキーワードがあります。

lr-tb	横書き（左から右、上から下へ）
tb-rl	縦書き（上から下、右から左へ）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>縦書きで表示したい</title>
<style type="text/css">
<!--
body          { line-height: 140% }
p#sample1     { writing-mode: lr-tb }
p#sample2     { writing-mode: tb-rl }
-->
</style>
</head>
<body>
<p id="sample1">
スタイルシートは、ひとことで……（中略）……もとに生み出されました。
</p>
```

```
<hr>
<p id="sample2">
スタイルシートは、ひとことで…… (中略) ……もとに生み出されました。
</p>
</body>
</html>
```


▲Netscapeは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
lr-tb	×	×	○	○	×	×	×
tb-rl	×	×	○	○	×	×	×

※適用するセレクタによっても効果が変わります

参照 縦書きの下線（傍線）位置を指定したい··········p.102

16 TEXT

縦書きの下線（傍線）位置を指定したい

text-underline-position: ★

★……キーワード

ページを縦書きで表示するよう設定した場合の下線（傍線）の位置を指定します。

下線の表示位置（テキストの上か下か）を指定するプロパティとしてInternet Explorer 5.5が独自に拡張したプロパティで、text-decoration: underlineと併用することで、縦書きのページの傍線の位置を指定できるようになります。

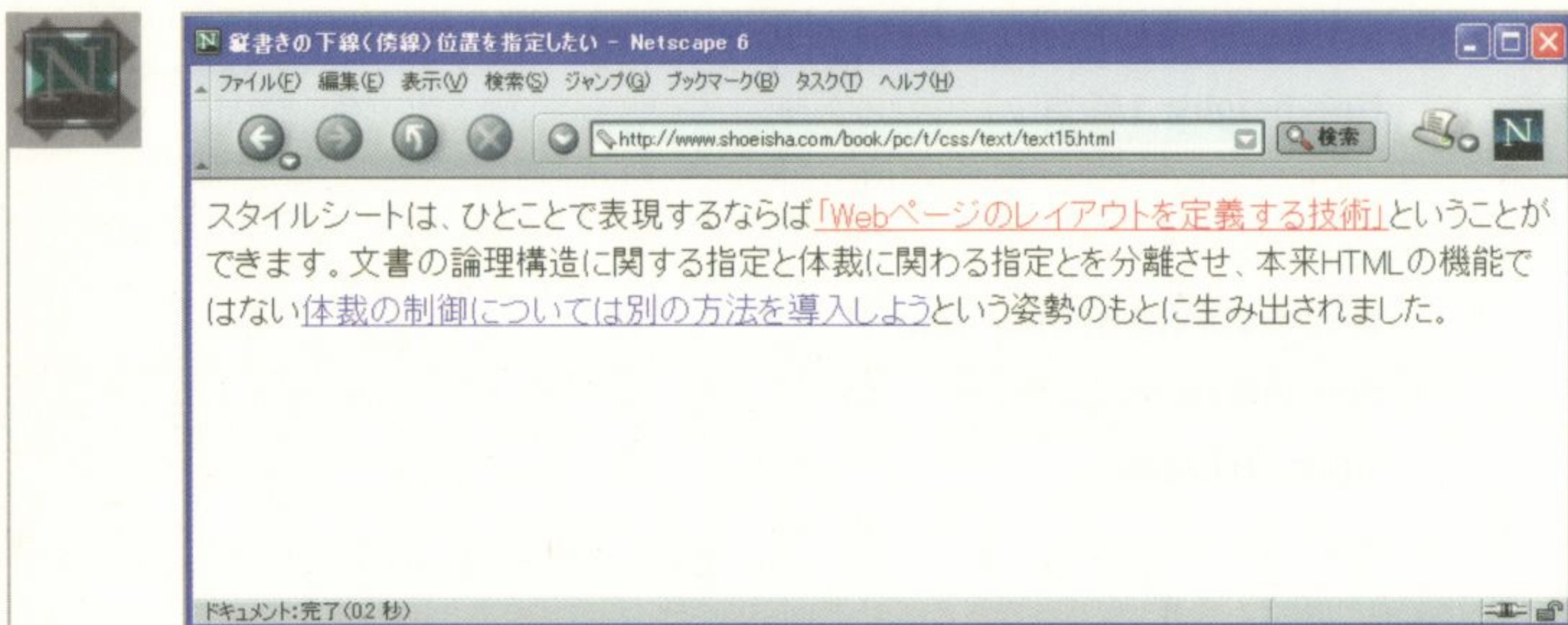
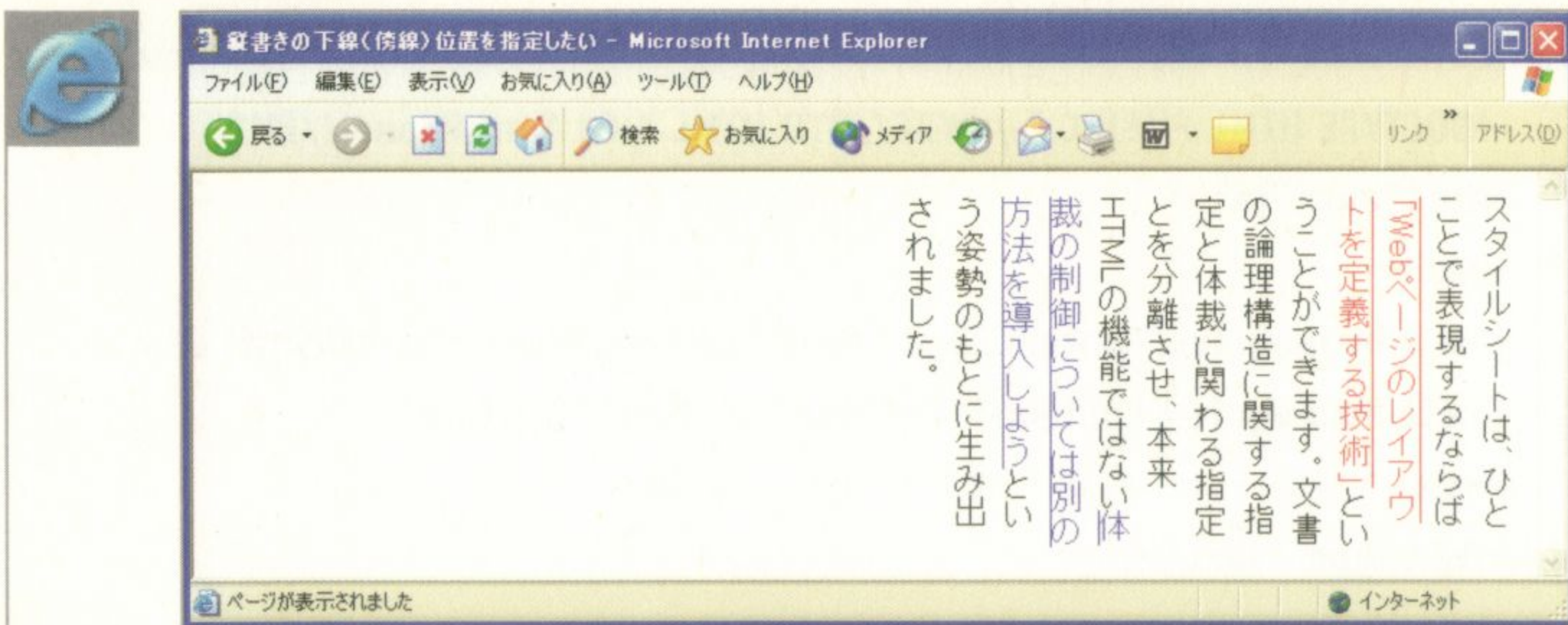
通常縦書きのテキストにtext-decoration: underlineで下線を指定すると下線は左につきますが、縦書きの場合の傍線はテキストの右側にあるほうが自然です。このような場合にtext-underline-position: aboveを同時に指定すると、傍線が文章の右側に表示されます。

なお、IE 5.5におけるデフォルトはbelowですが、IE 6ではautoがデフォルトに変更されています。

値には以下のキーワードがあります。

below	テキストの下。縦書きの場合にテキストの左側に下線を表示（IE 5.5のデフォルト）
above	テキストの上。縦書きの場合に文章の右側に下線を表示
auto	言語コードが日本語で縦書きの場合はテキストの上（＝右側）に下線を表示（IE 6以降のデフォルト）。それ以外はbelowとして処理
auto-pos	autoと同じ

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>縦書きの下線（傍線）位置を指定したい</title>
<style type="text/css">
<!--
p               {
        writing-mode: tb-rl;
        line-height: 140%
}
span#sample1    {
        text-decoration: underline;
        text-underline-position: auto;
        color: #ff0000
}
span#sample2    {
        text-decoration: underline;
        text-underline-position: below;
        color: #0000ff
}
-->
</style>
</head>
<body>
<p>
スタイルシートは、ひとことで表現するならば<span id="sample1">「Webページのレイアウトを定義する技術」</span>ということができます。文書の論理構造に関する指定と体裁に関わる指定とを分離させ、本来HTMLの機能ではない<span id="sample2">体裁の制御については別の方法を導入しよう</span>という姿勢のもとに生み出されました。
</p>
</body>
</html>
```

▲Netscapeは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
below	×	×	○	○	×	×	×
above	×	×	○	○	×	×	×
auto	×	×	×	○	×	×	×
auto-pos	×	×	×	○	×	×	×

※適用するセレクタによっても効果が変わります

参照 縦書きで表示したい……………………………p.100

ルビの配置を指定したい

ruby-align: ★
ruby-position: ★

★……キーワード

ruby-alignプロパティはルビの開始位置や文字間隔を、ruby-positionプロパティはルビが表示される位置を指定します。ruby要素にのみ適用します。

ruby-align

ruby-alignの値には以下のキーワードがあります。

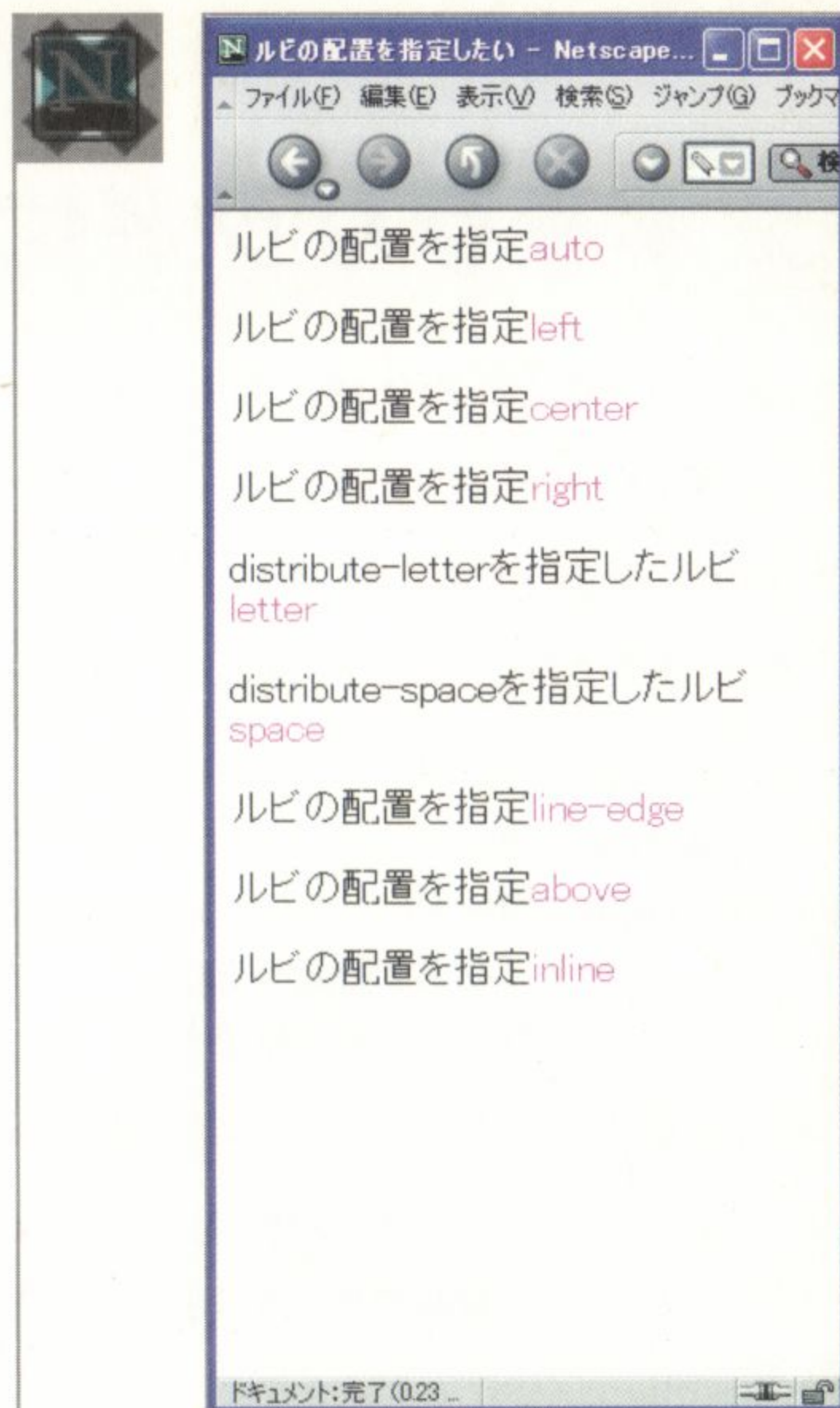
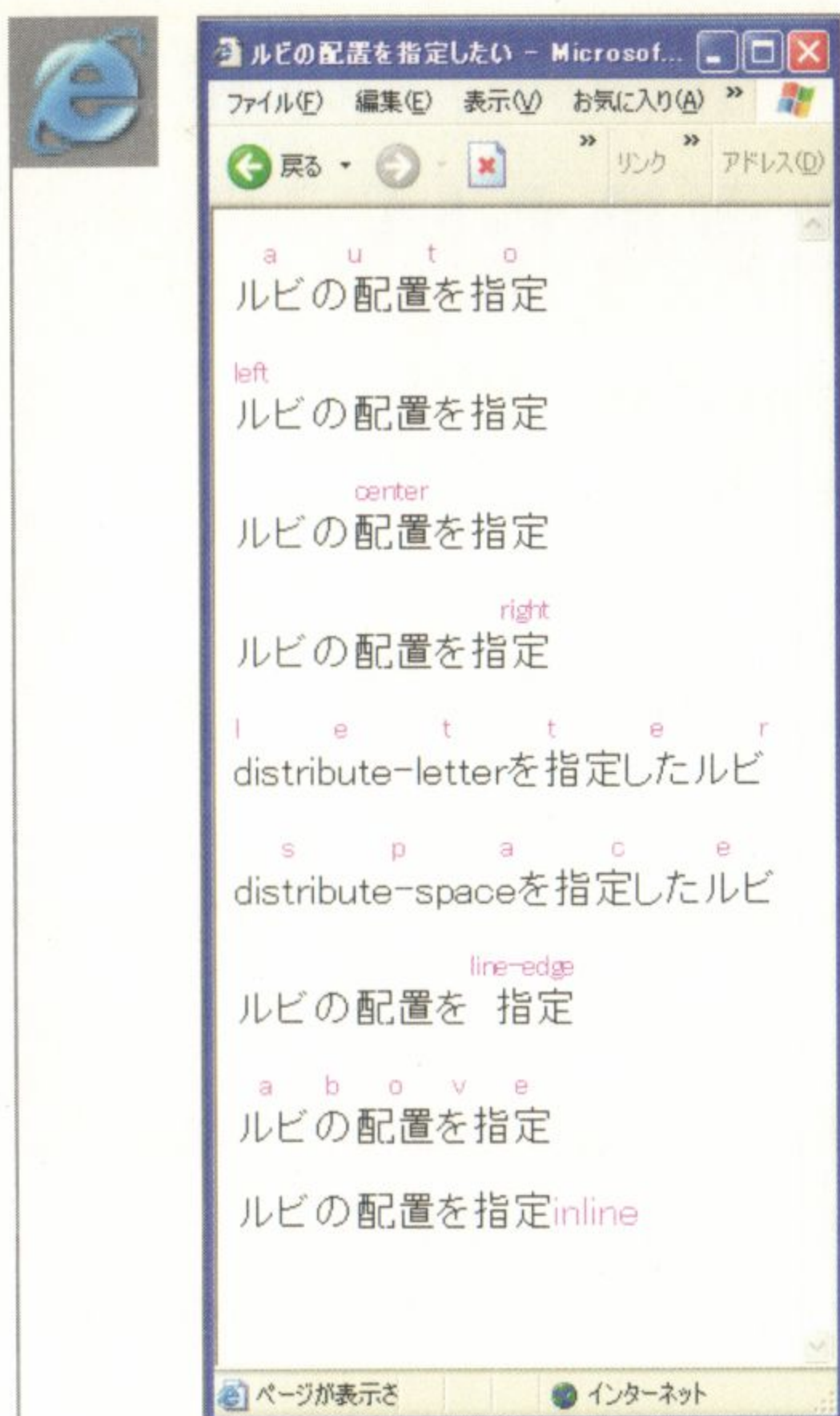
auto	ルビを自動的に割りふる（デフォルト）。日本語、中国語、韓国語などの場合はdistribute-space、アルファベットの場合はcenterが適用される
left	ルビをふるテキストに対して左詰めで表示する
center	ルビをふるテキストに対してセンタリングして表示する
rigt	ルビをふるテキストに対して右詰めで表示する
distribute-letter	ルビを均等に割りふる。ルビのほうが長い場合にはセンタリングして表示する
distribute-space	前後にスペースをあけてルビを均等に割りふる。ルビのほうが長い場合にはセンタリングして表示する
line-edge	ルビをふるテキストが行端にある場合はその端に寄せて表示する。それ以外はセンタリングして表示する

ruby-position

ruby-positionの値には以下のキーワードがあります。

above	ルビをふるテキストの上に表示する（デフォルト）
inline	ルビをふるテキストの直後に表示する

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>ルビの配置を指定したい</title>
<style type="text/css">
<!--
rt              { color: #ff00ff }
ruby.sample1  { ruby-align: auto }
ruby.sample2  { ruby-align: left }
ruby.sample3  { ruby-align: center }
ruby.sample4  { ruby-align: right }
ruby.sample5  { ruby-align: distribute-letter }
ruby.sample6  { ruby-align: distribute-space }
ruby.sample7  { ruby-align: line-edge }
ruby.sample8  { ruby-position: above }
ruby.sample9  { ruby-position: inline }
-->
</style>
</head>
<body>
<p><ruby class="sample1">ルビの配置を指定<rt>auto</ruby></p>
<p><ruby class="sample2">ルビの配置を指定<rt>left</ruby></p>
<p><ruby class="sample3">ルビの配置を指定<rt>center</ruby></p>
<p><ruby class="sample4">ルビの配置を指定<rt>right</ruby></p>
<p><ruby class="sample5">distribute-letterを指定したルビ<rt>letter</ruby></p>
<p><ruby class="sample6">distribute-spaceを指定したルビ<rt>space</ruby></p>
<p>ルビの配置を<ruby class="sample7">指定<rt>line-edge</ruby></p>
<p><ruby class="sample8">ルビの配置を指定<rt>above</ruby></p>
<p><ruby class="sample9">ルビの配置を指定<rt>inline</ruby></p>
</body>
</html>
```

▲Netscapeは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
auto	×	○	○	○	×	×	×
left	×	○	○	○	×	×	×
center	×	○	○	○	×	×	×
right	×	○	○	○	×	×	×
distribute-letter	×	○	○	○	×	×	×
distribute-space	×	○	○	○	×	×	×
line-edge	×	○	○	○	×	×	×
above	×	○	○	○	×	×	×
inline	×	○	○	○	×	×	×

※Macintosh版IE5はline-edgeに対応していません

18 TEXT

文字をグリッドにおさめたい

layout-grid-mode: ★	グリッドに合わせる方向
layout-grid-type: ☆	グリッドに合わせる方法
layout-grid-line: ▲	行送り
layout-grid-char: ▲	字送り

★••••••キーワード
☆••••••キーワード
▲••••••キーワード
サイズを表す数値＋単位
パーセントを表す数値＋％

縦横のグリッド（マス目）を想定し、その中に文字を当てはめるよう指定します。

なお、フォントの種類やサイズによってはグリッドから多少ずれることもありますので注意してください。

layout-grid-mode

layout-grid-modeは、グリッドに合わせる方向を指定します。

both	行方向・文字方向両方のグリッドに合わせる
none	グリッドを無効にする
line	行方向のみグリッドに合わせる
char	文字方向のみグリッドに合わせる

layout-grid-type

layout-grid-typeは、グリッドに合わせる方法を指定します。

loose	全角文字や半角カナを、layout-grid-charで指定されたグリッドの幅に合わせて間をあけて配置する（デフォルト）
fixed	すべての文字をグリッドの中心に配置する
strict	全角文字や半角カナのみをグリッドの中心に配置する

layout-grid-line、layout-grid-char

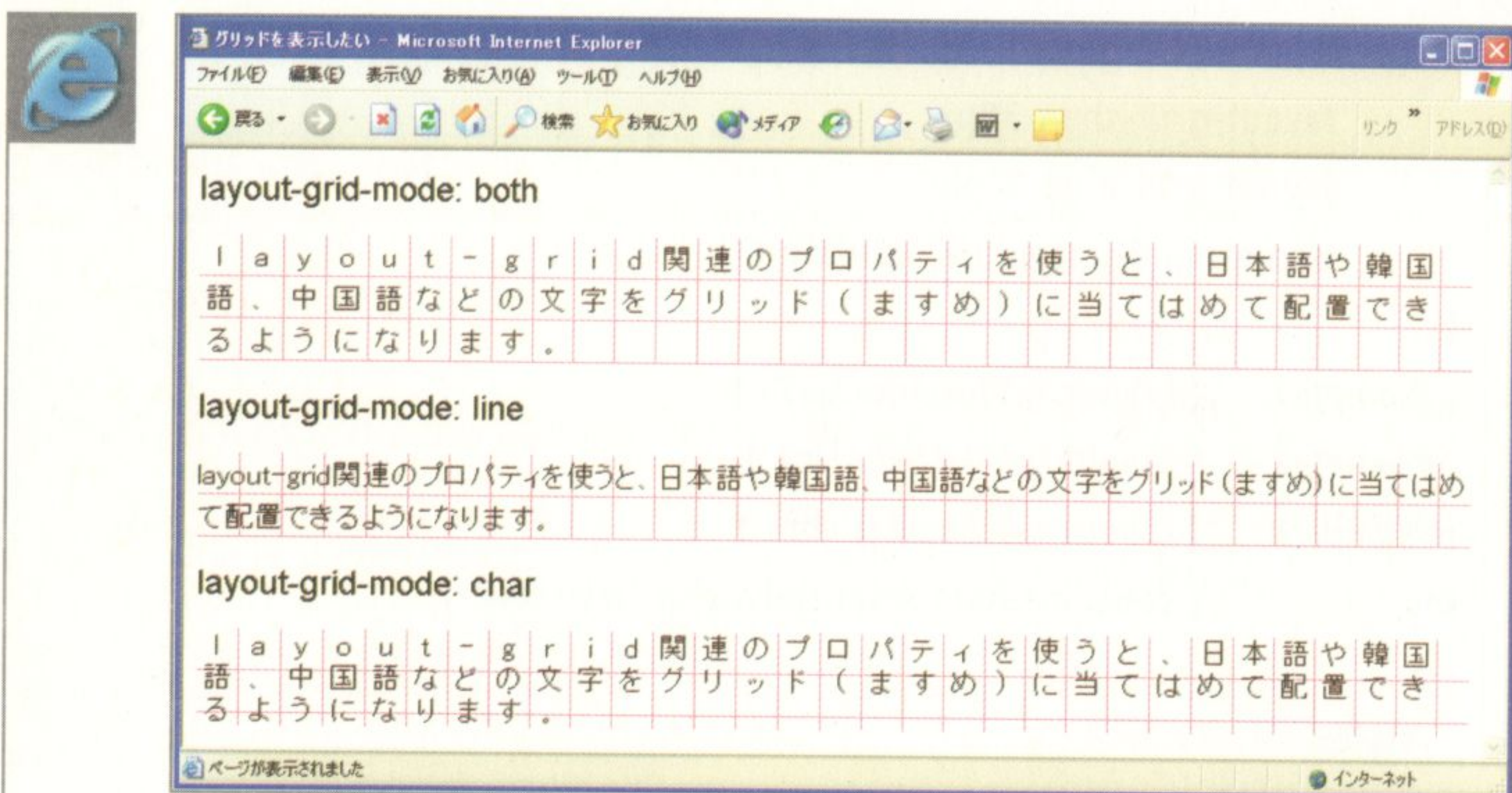
layout-grid-lineは行送り（行間）、layout-grid-charは字送りの幅（文字間）を指定します。

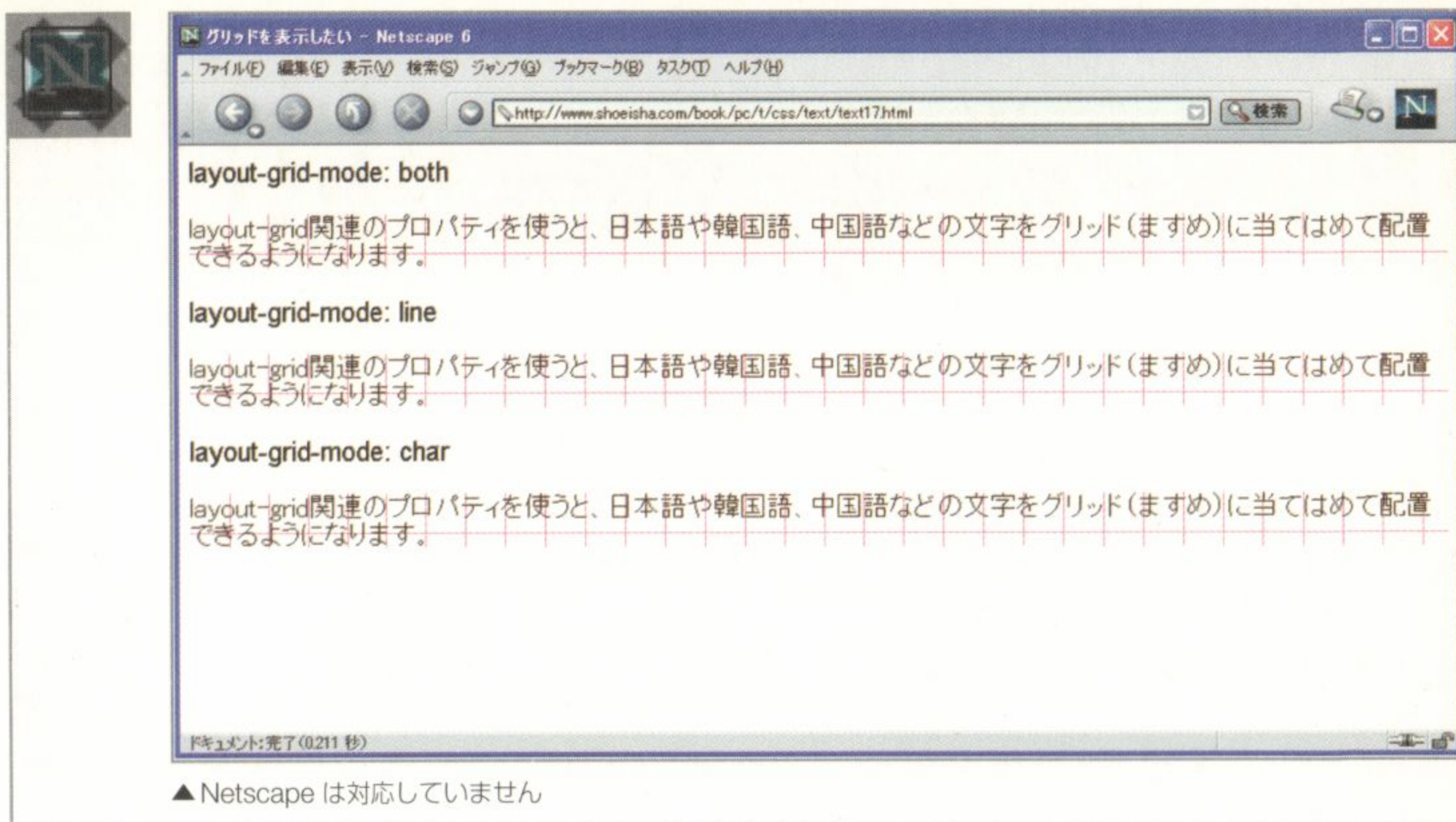
layout-grid-charやlayout-grid-line を有効にするには、layout-grid-modeでlineかbothが指定されている必要があります。

none	グリッドを設定しない（デフォルト）
auto	要素中で最大のフォントサイズをグリッドのサイズに設定する
サイズを表す数値＋単位	数値に単位をつけてグリッドのサイズを設定する。単位についてはp.46を参照してください
パーセントを表す数値＋％	親要素のボックス領域のサイズに対する割合で設定する

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>グリッドを表示したい</title>
<style type="text/css">
<!--
p            {
        layout-grid-type: fixed;
        layout-grid-char: 30px;
        layout-grid-line: 30px;
        background-image:url("grid1.gif")
}
p#sample1    { layout-grid-mode: both }
p#sample2    { layout-grid-mode: line }
p#sample3    { layout-grid-mode: char }
div          { font: medium Arial,Helvetica,sans-serif }
-->
</style>
</head>
<body>

<div>layout-grid-mode: both</div>
```

```
<p id="sample1">
layout-grid 関連のプロパティを使うと、日本語や韓国語、中国語などの文字をグリッド（ま
すめ）に当てはめて配置できるようになります。
</p>

<div>layout-grid-mode: line</div>
<p id="sample2">
layout-grid 関連のプロパティを使うと、日本語や韓国語、中国語などの文字をグリッド（ま
すめ）に当てはめて配置できるようになります。
</p>

<div>layout-grid-mode: char</div>
<p id="sample3">
layout-grid 関連のプロパティを使うと、日本語や韓国語、中国語などの文字をグリッド（ま
すめ）に当てはめて配置できるようになります。
</p>

</body>
</html>
```

▲layout-grid-mode で line を設定すると行送りが、char を設定すると字送りがグリッドに揃います（ここではわかりやすくするためにグリッド線の画像を背景に表示させています）

▲Netscapeは対応していません

layout-grid-mode	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
none	×	○	○	○	×	×	×
both	×	○	○	○	×	×	×
line	×	○	○	○	×	×	×
char	×	○	○	○	×	×	×

layout-grid-type	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
loose	×	○	○	○	×	×	×
fixed	×	○	○	○	×	×	×
strict	×	○	○	○	×	×	×

layout-grid-char	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
auto	×	○	○	○	×	×	×
サイズ	×	○	○	○	×	×	×
パーセント	×	○	○	○	×	×	×

layout-grid-line	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
auto	×	○	○	○	×	×	×
サイズ	×	○	○	○	×	×	×
パーセント	×	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります
※Macintosh版IE5は対応していません

グリッドを一括して指定したい……………………p.112

19 TEXT

グリッドを一括して指定したい

layout-grid: ★ ☆ ▲ △

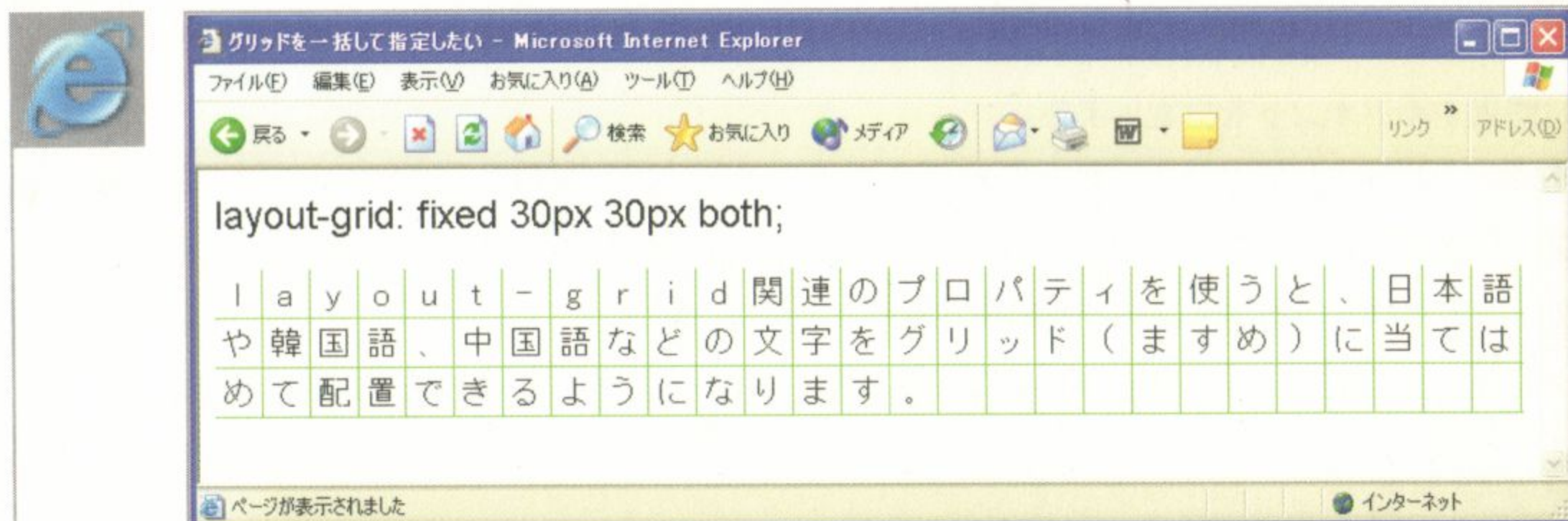
★••••••layout-grid-modeの値
☆••••••layout-grid-typeの値
▲••••••layout-grid-lineの値
△••••••layout-grid-charの値

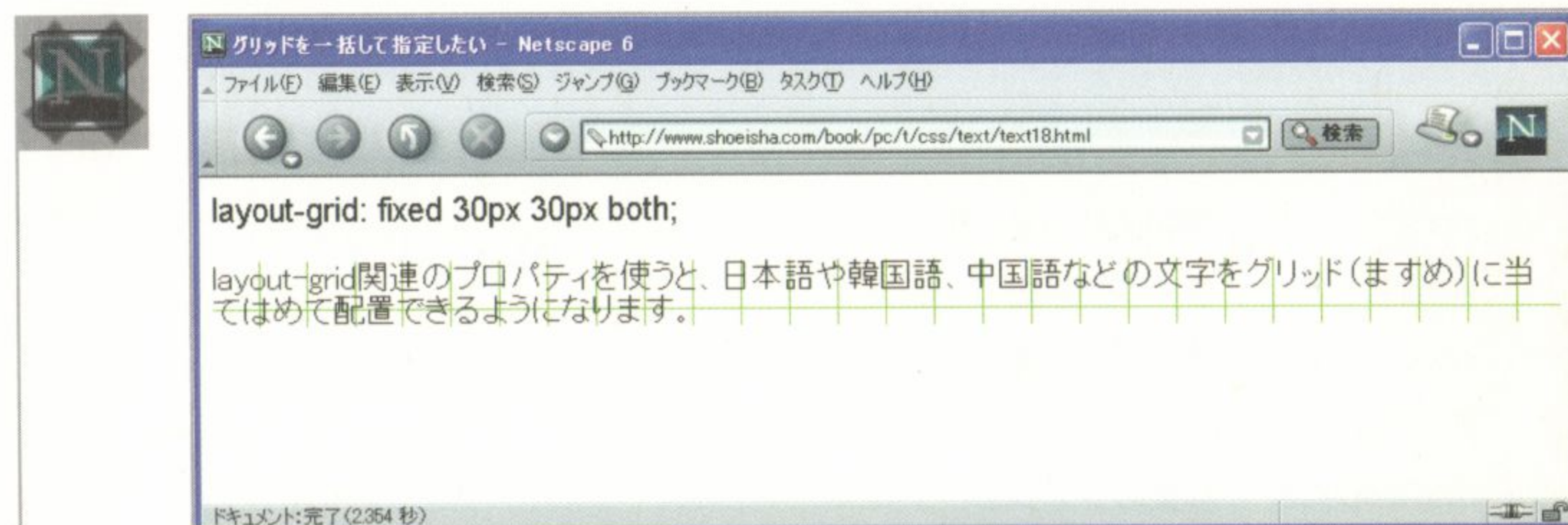
グリッドに関する各種指定を一括して設定します。

グリッドに合わせる方向（layout-grid-mode）、方法（layout-grid-type）、行送り（layout-grid-line）、字送り（layout-grid-char）のそれぞれの値を、半角スペースで区切って任意の順番で指定します。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>グリッドを一括して指定したい</title>
<style type="text/css">
<!--
p       {
        font-size: 14pt;
        layout-grid: fixed 30px 30px both;
        background-image: url("grid2.gif")
}
div     { font: medium Arial,Helvetica,sans-serif }
-->
</style>
</head>
```

```
<body>
<div>layout-grid: fixed 30px 30px both</div>
<p>
layout-grid 関連のプロパティを使うと、日本語や韓国語、中国語などの文字をグリッド（ますめ）に当てはめて配置できるようになります。
</p>
</body>
</html>
```


▲ここではわかりやすくするためにグリッド線の画像を背景に表示させています

▲ Netscape は対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
共通	×	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります

※ Macintosh 版 IE5 は対応していません

文字をグリッドにおさめたい……………………p.108

1 FONT

フォントを指定したい

フォント

font-family: ★

★……フォント名
　　キーワード

使用するフォントを指定します。

複数の候補を指定するときは、それぞれを「,」（カンマ）で区切って指定します。その場合は並べた順に優先順位がつき、先に指定されているフォントがユーザー側の環境にない場合には、次に指定されているフォントで表示を試みるようになります。

フォントの指定には、フォント名による方法と、キーワードによる方法とがあります。

フォント名

フォントを具体的な名前で指定します。フォントの名前は文字の全角や半角、スペースなども含めて正しく記述してください。フォント名にスペースが含まれている場合は引用符（「"」または「'」）で囲む必要があります。

キーワード

serif	明朝系のフォント
sans-serif	ゴシック系のフォント
cursive	筆記体・草書体系のフォント
fantasy	装飾がメインとなっているフォント
monospace	等幅のフォント

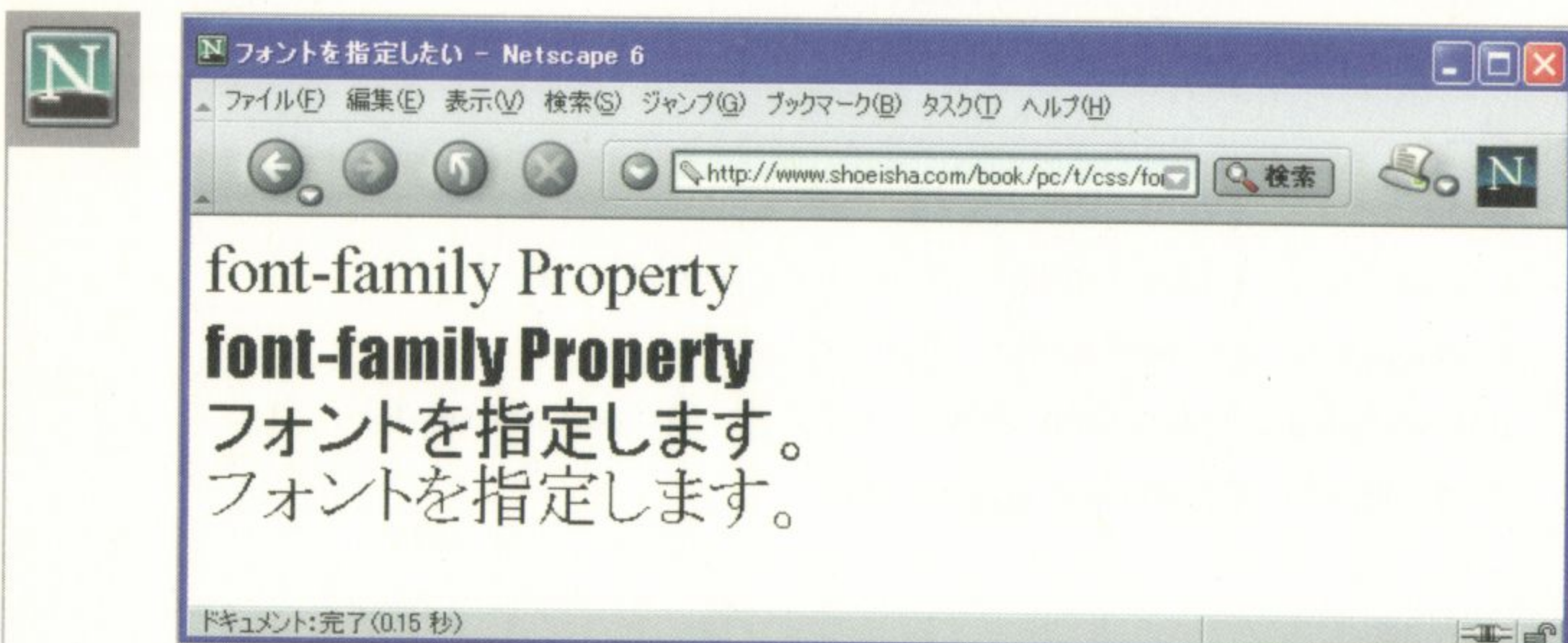
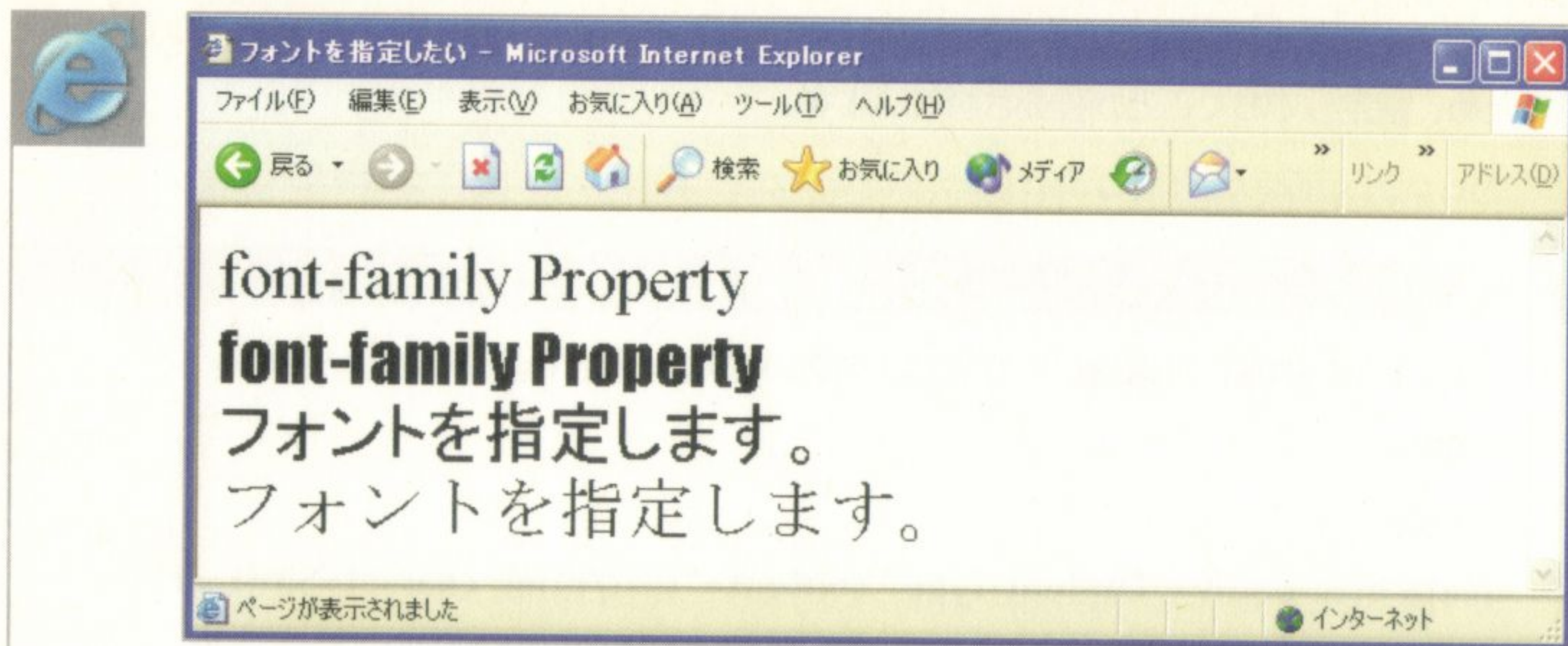
これらのキーワードはフォントの種類を表します。良く似た特徴を持つフォントをカテゴリ分けしたもので、実際に表示する場合にはブラウザによってカテゴリ内からフォントが１種類選択されます。それぞれのキーワードに属するフォントの例は下表を参照してください。

serif	Times New Roman	Garamond	MS Georgia	太ミン	リュウミン	MS明朝
sans-serif	Helvetica	MS Arial	MS Verdana	平成角ゴシック	新ゴシック	MSゴシック
cursive	Caflisch Script (caflisch script)	Ex Ponto (ex ponto)				
fantasy	CRITTER (critter)	STUDZ (studz)				
monospace	Courier	MS Courier New	Osaka-等幅			

▲各キーワードに含まれるフォントの例

キーワードによる指定は、フォント名による指定がどれも有効でない場合の最終的な選択肢になるため、指定しておくことが推奨されています。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>フォントを指定したい</title>
<style type="text/css">
<!--
body          { font-size: 25pt }
div#sample1   { font-family: "Times New Roman",serif }
div#sample2   { font-family: Impact,sans-serif }
div#sample3   { font-family: 'ＭＳ Ｐゴシック',Osaka,sans-serif }
div#sample4   { font-family: serif }
-->
</style>
</head>
<body>
<div id="sample1">font-family Property</div>
<div id="sample2">font-family Property</div>
<div id="sample3">フォントを指定します。</div>
<div id="sample4">フォントを指定します。</div>
</body>
</html>
```

フォント

フォントを指定するHTMLタグをCSSに改める

HTMLタグでフォントの種類を指定するには、次のようにタグのface属性を利用します。

～　　　　（★──フォント名）

<font>タグはDeprecated（推奨しない）とされており、表示フォントの種類はスタイルシートで指定することが推奨されています。同様の効果をスタイルシートで表現すると以下のようになります。

☆ { font-family: ★,★,...}　　　　（☆──セレクタ　★──フォント名）

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
フォント名	○	○	○	○	△＊1	△＊1	○
serif	○	○	○	○	△＊1	△＊1	○
sans-serif	○	○	○	○	×	×	○
cursive	○	○	○	○	×	×	○
fantasy	○	○	○	○	×	×	○
monospace	○	○	○	○	△＊1	△＊1	○

＊1：［文字コードセット］が日本語になっていると正常に動作しません
※適用するセレクタによっても効果が変わります

参照
フォントサイズを指定したい…………………p.117
フォントの太さを指定したい…………………p.122
フォントを一括して指定したい………………p.130

フォントサイズを指定したい

font-size: ★

★……サイズを表す数値＋単位
パーセントを表す数値＋%
キーワード
キーワード（相対的な指定）

フォントの大きさを指定します。

値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけてフォントサイズを指定します。単位についてはp.46を参照してください。

パーセントを表す数値＋%

親要素のフォントサイズに対する割合で指定します。たとえば親要素が10ptの場合に120%を指定すると12ptになります。

キーワード

xx-small	非常に小さい
x-small	小さい
small	やや小さい
medium	通常のサイズ（デフォルト）
large	やや大きい
x-large	大きい
xx-large	非常に大きい

フォントサイズを7種類のキーワードで指定します。xx-smallでもっとも小さく、xx-largeでもっとも大きなサイズに設定されます。CSS1の定義では1.5倍ずつ大きくなるという定義でしたが、CSS2では1.2倍に変更されています。実際の表示はブラウザやフォントの種類によって異なることもあります。

なお、CSS2ではフォントサイズのデフォルトとしてmediumが定義されていますが、Internet Explorerで標準準拠モードが有効になっている（p.51参照）場合、デフォルトのフォントサイズはmediumではなくsmallになります。

キーワード（相対的な指定）

larger　親要素に対して一段階大きく
smaller　親要素に対して一段階小さく

親要素のフォントサイズに対する相対的な大きさを指定します。これらのキーワードを設定すると、親要素のフォントサイズに従って前述の７種類のキーワードを解釈し、フォントサイズを設定します。たとえば親要素のフォントサイズがmediumの場合にlarger（一段階大きく）を指定すると、フォントサイズはlargeになります。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>フォントサイズを指定したい</title>
<style type="text/css">
<!--
span            { margin:10pt 5pt }
span#sample1    { font-size: 15pt }
span#sample2    { font-size: 2em }
span#sample3    { font-size: 120% }
span#sample4    { font-size: 75% }
span#sample5    { font-size: xx-small }
span#sample6    { font-size: x-small }
span#sample7    { font-size: small }
span#sample8    { font-size: medium }
span#sample9    { font-size: large }
span#sample10   { font-size: x-large }
span#sample11   { font-size: xx-large }
span#sample12   { font-size: larger }
span#sample13   { font-size: smaller }
span.initial    { color: #ff0000 }
-->
</style>
</head>
```

```
<body>
<p>
<span class="initial">初期値（default）</span>
<span id="sample1">15ptを指定</span>
<span id="sample2">2emを指定</span>
</p>
<p>
<span class="initial">初期値（default）</span>
<span id="sample3">120%を指定</span>
<span id="sample4">75%を指定</span>
</p>
<p>
<span id="sample5">xx-small</span><br>
<span id="sample6">x-small</span><br>
<span id="sample7">small</span><br>
<span id="sample8">medium</span><br>
<span id="sample9">large</span><br>
<span id="sample10">x-large</span><br>
<span id="sample11">xx-large</span>
</p>
<p>
<span class="initial">初期値（default）</span>
<span id="sample12">largerを指定</span>
<span id="sample13">smallerを指定</span>
</p>
</body>
</html>
```

フォント

フォントサイズを指定したい - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る 検索 お気に入り メディア
リンク アドレス(D)
初期値(default) 15ptを指定 2emを指定
初期値(default) 120%を指定 75%を指定
xx-small
x-small
small
medium
large
x-large
xx-large
初期値(default) largerを指定 smallerを指定
ページが表示されました
インターネット

フォントサイズを指定したい - Netscape 6
ファイル(F) 編集(E) 表示(V) 検索(S) ジャンプ(G) ブックマーク(B) タスク(T) ヘルプ(H)
http://www.shoeisha.com/book/pc/t/css/font/font02.html
検索
初期値(default) 15ptを指定 2emを指定
初期値(default) 120%を指定 75%を指定
xx-small
x-small
small
medium
large
x-large
xx-large
初期値(default) largerを指定 smallerを指定
ドキュメント：完了(0.25 秒)

フォントサイズを指定するHTMLタグをCSSに改める

HTMLタグでフォントのサイズを指定するには、次のようにタグのsize属性を利用します。

～　　絶対的な指定（★──1～7）
<basefont size="▲">～
　　相対的な指定（▲──1～7　△──▲との±が7になる数値）
<big>～</big>　　大きめ
<small>～</small>　　小さめ

<font>タグはDeprecated（推奨しない）とされており、フォントサイズはスタイルシートで指定することが推奨されています。また、HTMLタグでは絶対的・相対的どちらの場合も、7段階のサイズ指定しかできませんが、スタイルシートを使うと詳細で柔軟なサイズ指定が可能です。同様の効果をスタイルシートで表現すると以下のようになります。

◆ { font-size: ◇ }　　◇の単位によって絶対的・相対的のどちらも可能（◇──サイズ）
◆ { font-size: larger }　　大きめ
◆ { font-size: smaller }　　小さめ
◆──セレクタ

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○
キーワード	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

行の高さを設定したい……………………………p.66
フォントの太さを指定したい……………………p.122
フォントを一括して指定したい………………p.130

フォントの太さを指定したい

font-weight: ★

★・・・・・・キーワード
数値

フォントの太さを指定します。
値には次のような指定方法があります。

数値

100,200,300,400,500,600,700,800,900

標準の大きさを400とし、数値が大きくなればより大きなフォント、数値が小さけくなればより小さなフォントに設定されます。たいていのフォントはこれらの9種類の太さをすべて揃えているわけではなく、またブラウザによっても太さの解釈が異なります。そのため指定した値によっては太さが変わらないこともあります。

Internet Explorerにおける主なフォントの対応状況は以下のとおりです。

指定する数値	100	200	300	400	500	600	700	800	900
MS明朝			400				600		
MSゴシック			400				600		
Arail			400			600	700		900
Bookman Old Style			400		500	600		800	
Comic Suns MS			400			600	700		900
Courier			400			600	700		900
Garamond			400			600	700		900
Times New Roman			400			600	700		900

▲実際に表示されるフォントサイズ

キーワード

normal　標準の大きさ（デフォルト）
bold　太字
bolder　一段階太く
lighter　一段階細く

boldを指定すると、太字になり、700を指定した場合と同じ結果になります。

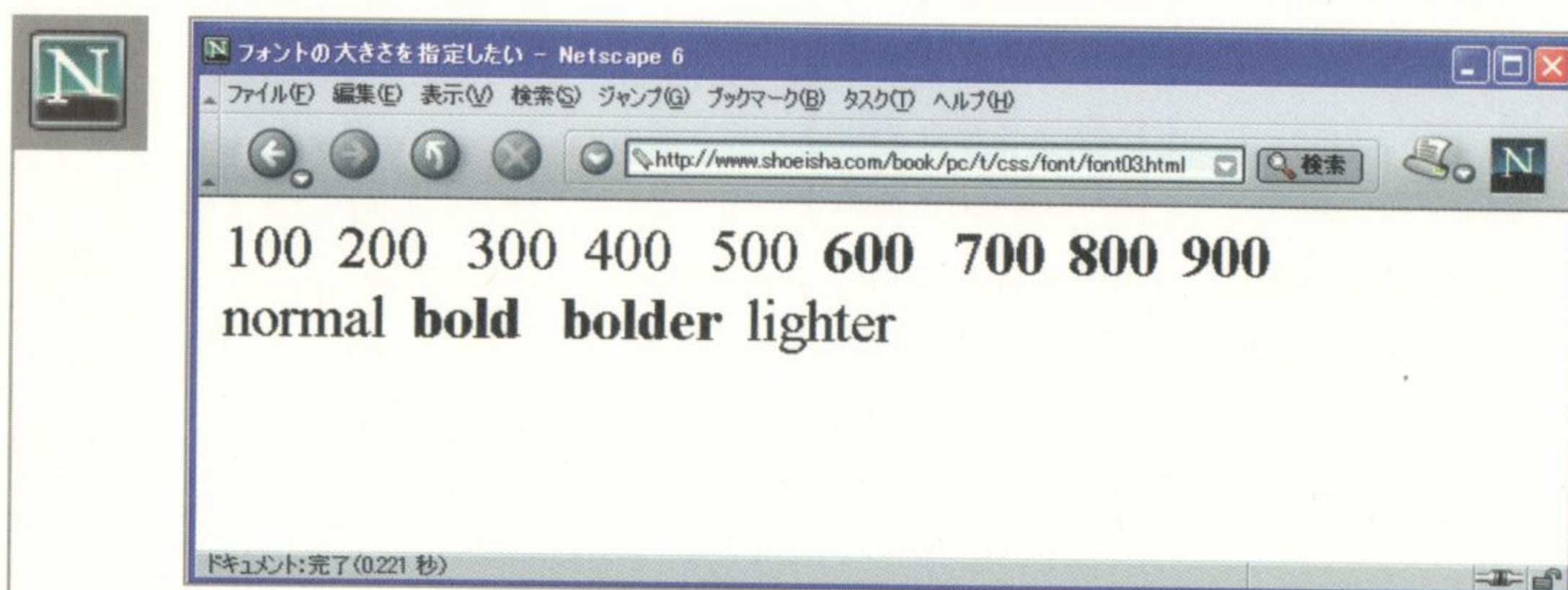
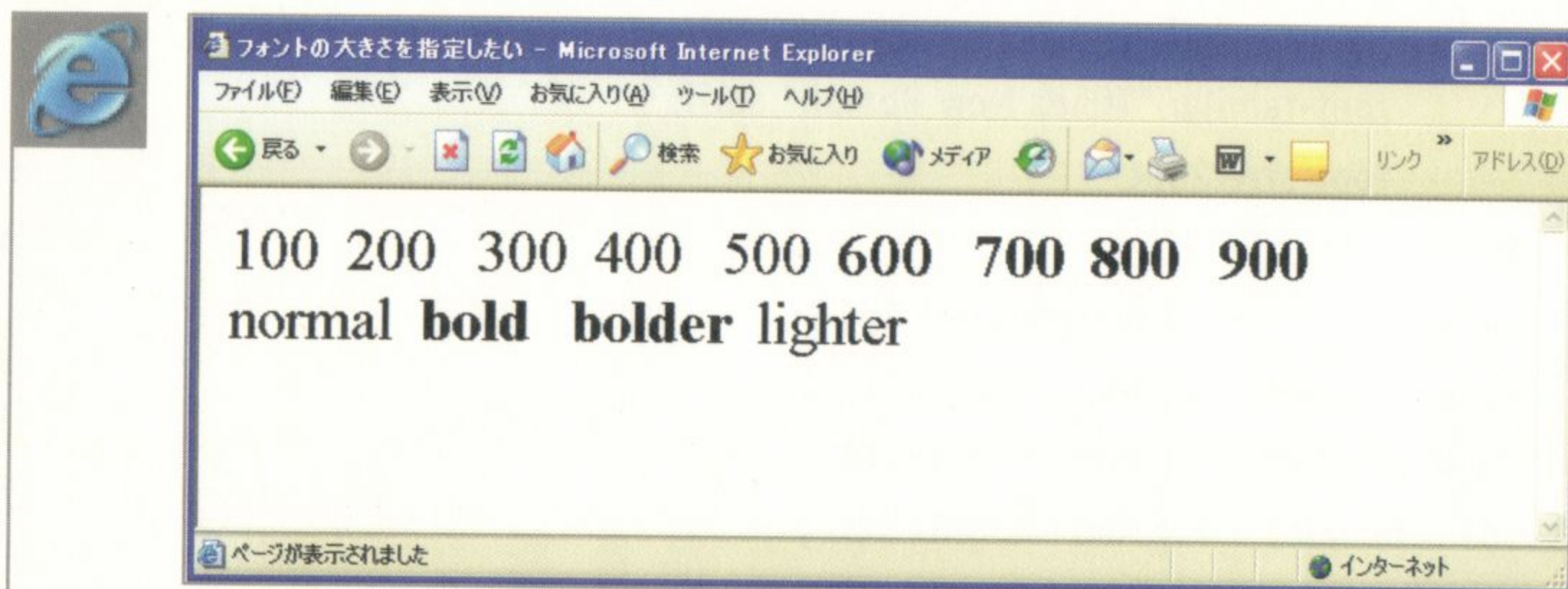
bolderとlighterではその時点での太さよりも一段階太く、あるいは一段階細くなります。指

定された値の太さが存在しない場合にはプロパティの値は変更されますが、表示される太さは変わりません。また、900が設定されている場合により大きな値を指定することはできませんし、100が設定されている場合により小さな値を指定することもできません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>フォントの大きさを指定したい</title>
<style type="text/css">
<!--
body            {
        font-family: "Times New Roman",serif;
        font-size:25pt
}
span            { margin:10pt 5pt }
span#sample1    { font-weight: 100 }
span#sample2    { font-weight: 200 }
span#sample3    { font-weight: 300 }
span#sample4    { font-weight: 400 }
span#sample5    { font-weight: 500 }
span#sample6    { font-weight: 600 }
span#sample7    { font-weight: 700 }
span#sample8    { font-weight: 800 }
span#sample9    { font-weight: 900 }
span#sample10   { font-weight: bold }
span#sample11   { font-weight: bolder }
span#sample12   { font-weight: lighter }
-->
</style>
</head>
<body>
<div>
```

```
<span id="sample1">100</span><span id="sample2">200</span>
<span id="sample3">300</span><span id="sample4">400</span>
<span id="sample5">500</span><span id="sample6">600</span>
<span id="sample7">700</span><span id="sample8">800</span>
<span id="sample9">900</span>
</div>
<div>
<span>normal</span><span id="sample10">bold</span>
<span id="sample11">bolder</span><span id="sample12">lighter</span>
</div>
</body>
</html>
```

フォントの太さを指定するHTMLタグをCSSに改める

HTMLタグで太字フォントを指定するには、タグを使用します。

～

HTMLタグでフォントの太さを指定するには<b>タグしかありませんが（<strong>など論理的定義の結果としてブラウザ側で太字表示するタグは除く）、スタイルシートなら細字にすることもできるほか、太さの度合いについても細かく指定することができます。同様の効果をスタイルシートで表現すると以下のようになります。

★ { font-weight: bold }　　　　（★──セレクタ）

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
100～900	×	○	○	○	○	○	○
normal	○	○	○	○	○	○	○
bold	○	○	○	○	○	○	○
bolder	×	○	○	○	○	○	○
lighter	×	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

フォントサイズを指定したい……………………p.117
フォントを一括して指定したい………………p.130

フォントを斜体にしたい

font-style: ★

★……キーワード

文字を斜体にします。

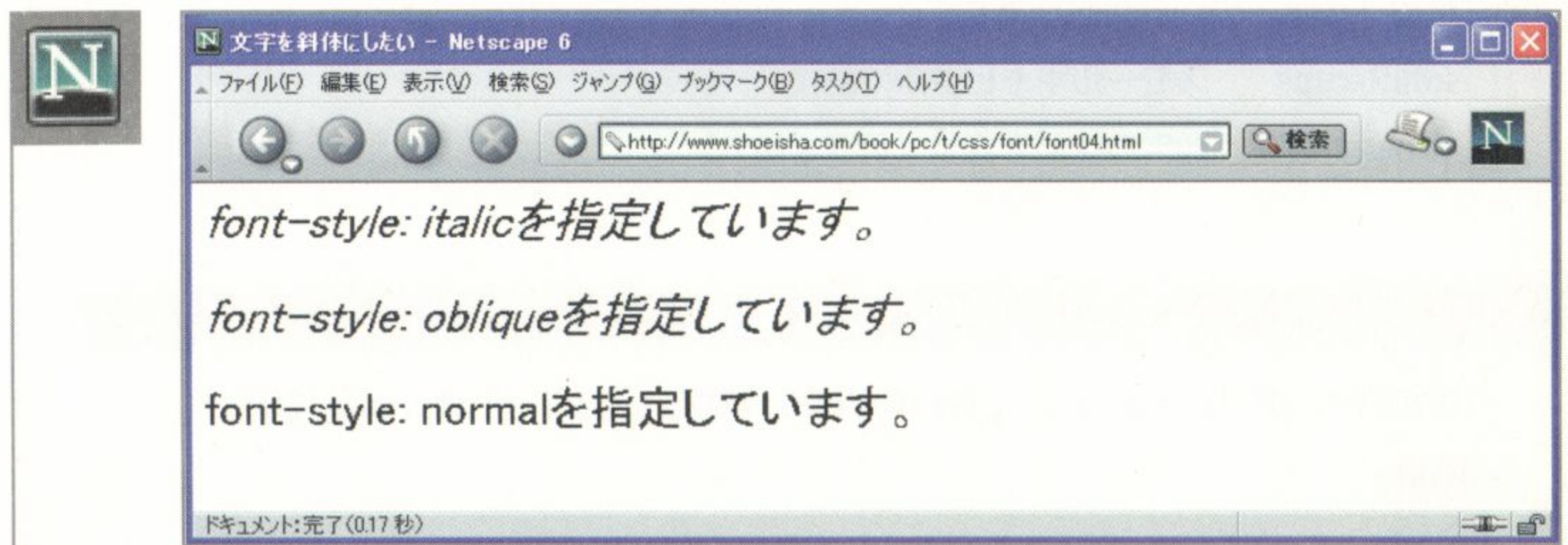
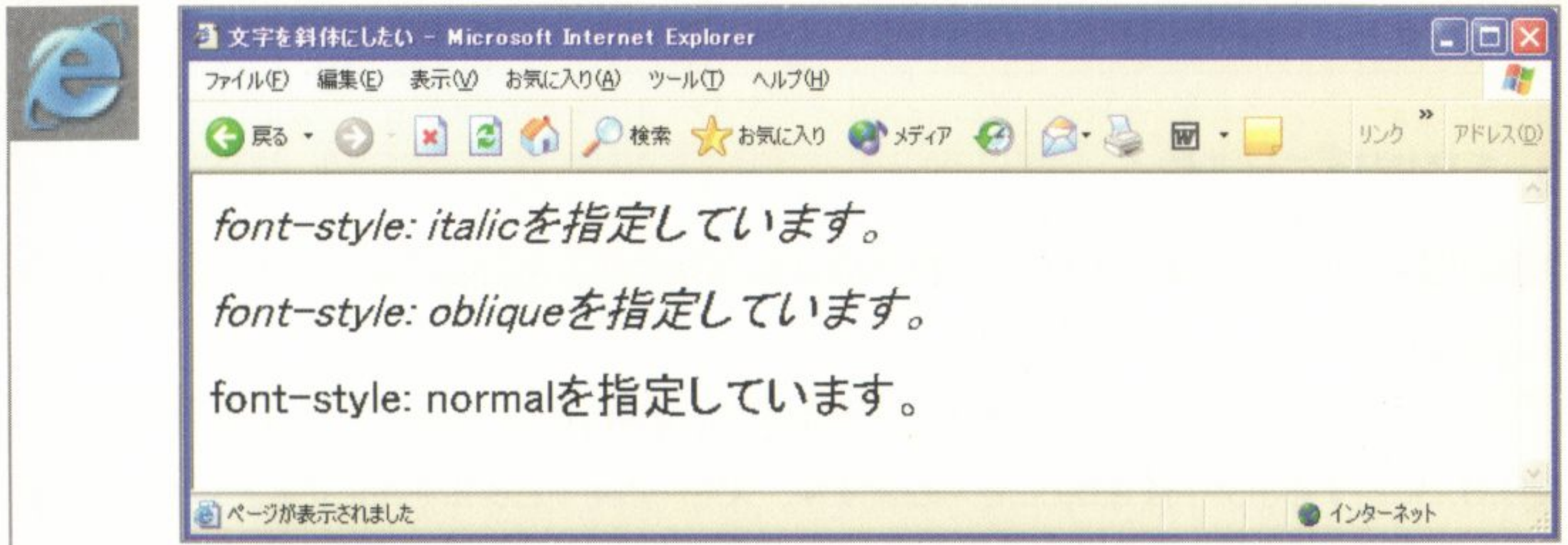
値には以下のキーワードがあります。

italic	斜体
oblique	斜体
normal	通常の状態で表示（デフォルト）

厳密な定義によればitalicとobliqueは異なります。しかし、現在のところ一般的なブラウザではその違いを区別しておらず、同じように表示されます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>文字を斜体にしたい</title>
<style type="text/css">
<!--
body        { font-size: 20pt }
p#sample1   { font-style: italic }
p#sample2   { font-style: oblique }
p#sample3   { font-style: normal }
-->
</style>
</head>
<body>
```

```
<p id="sample1">font-style: italic を指定しています。</p>
<p id="sample2">font-style: oblique を指定しています。</p>
<p id="sample3">font-style: normal を指定しています。</p>
</body>
</html>
```


斜体を指定するHTMLタグをCSSに改める

HTMLタグで斜体を指定するには、<i>タグを利用します。

<i>～</i>

同様の効果をスタイルシートで表現すると以下のようになります。

★ { font-style: ☆ }　　（★──セレクタ　☆──italic または oblique）

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
italic	○	○	○	○	○	○	○
oblique	○	○	○	○	×	×	○
normal	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照
文字を装飾したい……………………………p.60
フォントを一括して指定したい………………p.130

5 FONT

文字にスモールキャピタルを指定したい

font-variant: ★

★……キーワード

文字をスモールキャピタルに指定します。

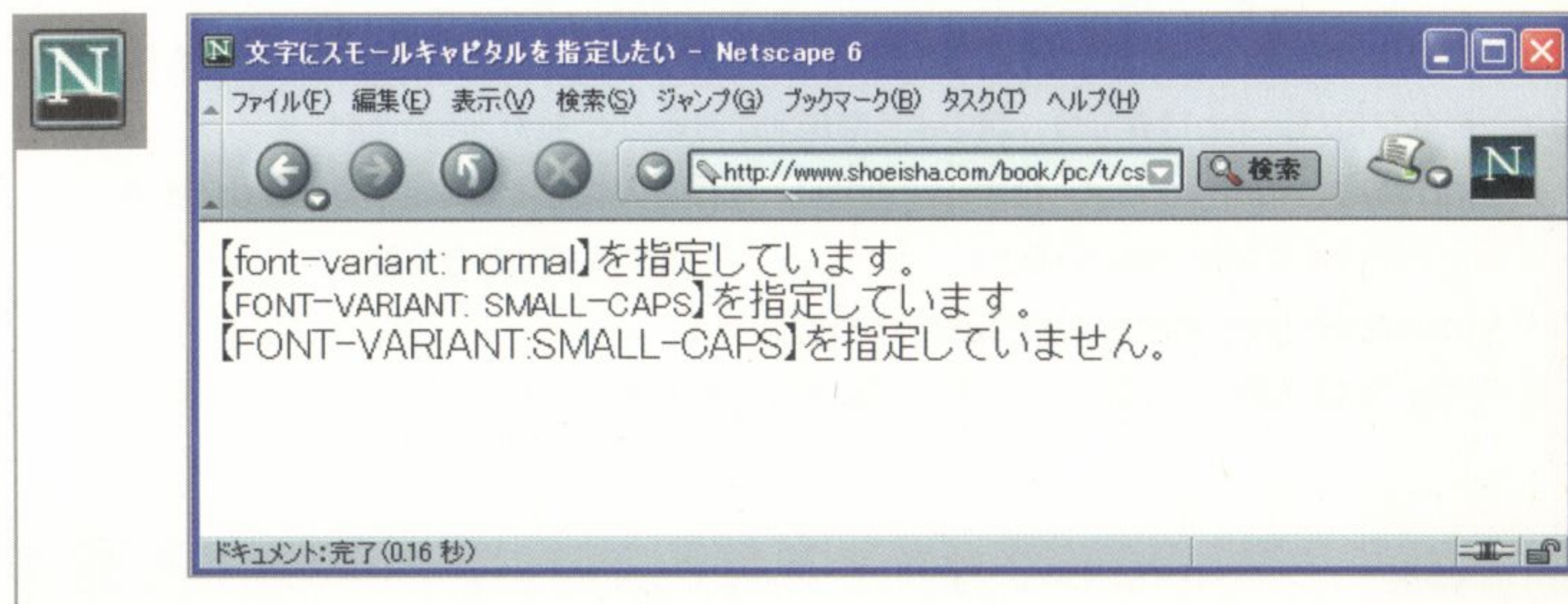
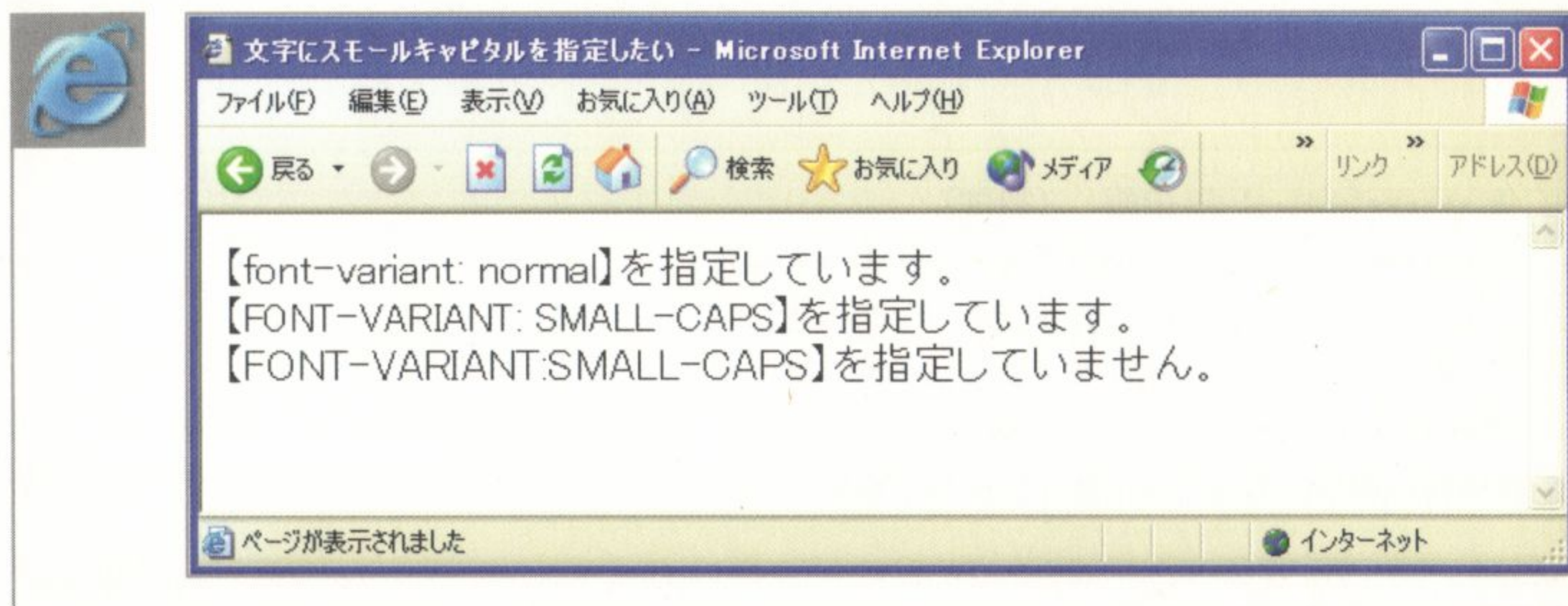
スモールキャピタルとは、大文字の字体を小さくした表示です。たとえば一般的には大文字A,B,C...に対する小文字はa,b,c...となりますが、スモールキャピタルを指定すると、小文字であってもA,B,C...とすべて大文字に変換され、かつ通常よりやや小さなサイズで表示されます。

値には以下のキーワードがあります。

small-caps	スモールキャピタルに置換
normal	通常の状態で表示（デフォルト）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>文字にスモールキャピタルを指定したい</title>
<style type="text/css">
<!--
div#sample1  { font-variant: normal }
div#sample2  { font-variant: small-caps }
-->
</style>
</head>
<body>
<div id="sample1">【font-variant: normal】を指定しています。</div>
<div id="sample2">【font-variant: small-caps】を指定しています。</div>
```

```
<div>【FONT-VARIANT:SMALL-CAPS】を指定していません。</div>
</body>
</html>
```


	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
small-caps	○	○	○	×	×	○	○
normal	○	○	○	×	×	○	○

※適用するセレクタによっても効果が変わります

参照
大文字・小文字に置換したい ……………………p.63
フォントを一括して指定したい…………………p.130

6 FONT

フォントを一括して指定したい

font: ★ ☆ ▲ △/▼ ▽

★••••••font-styleの値　（斜体）
☆••••••font-variantの値（スモールキャピタル）
▲••••••font-weightの値（フォントの太さ）
△••••••font-sizeの値　（フォントサイズ）
▼••••••line-heightの値（行の高さ）
▽••••••font-familyの値（フォント名）

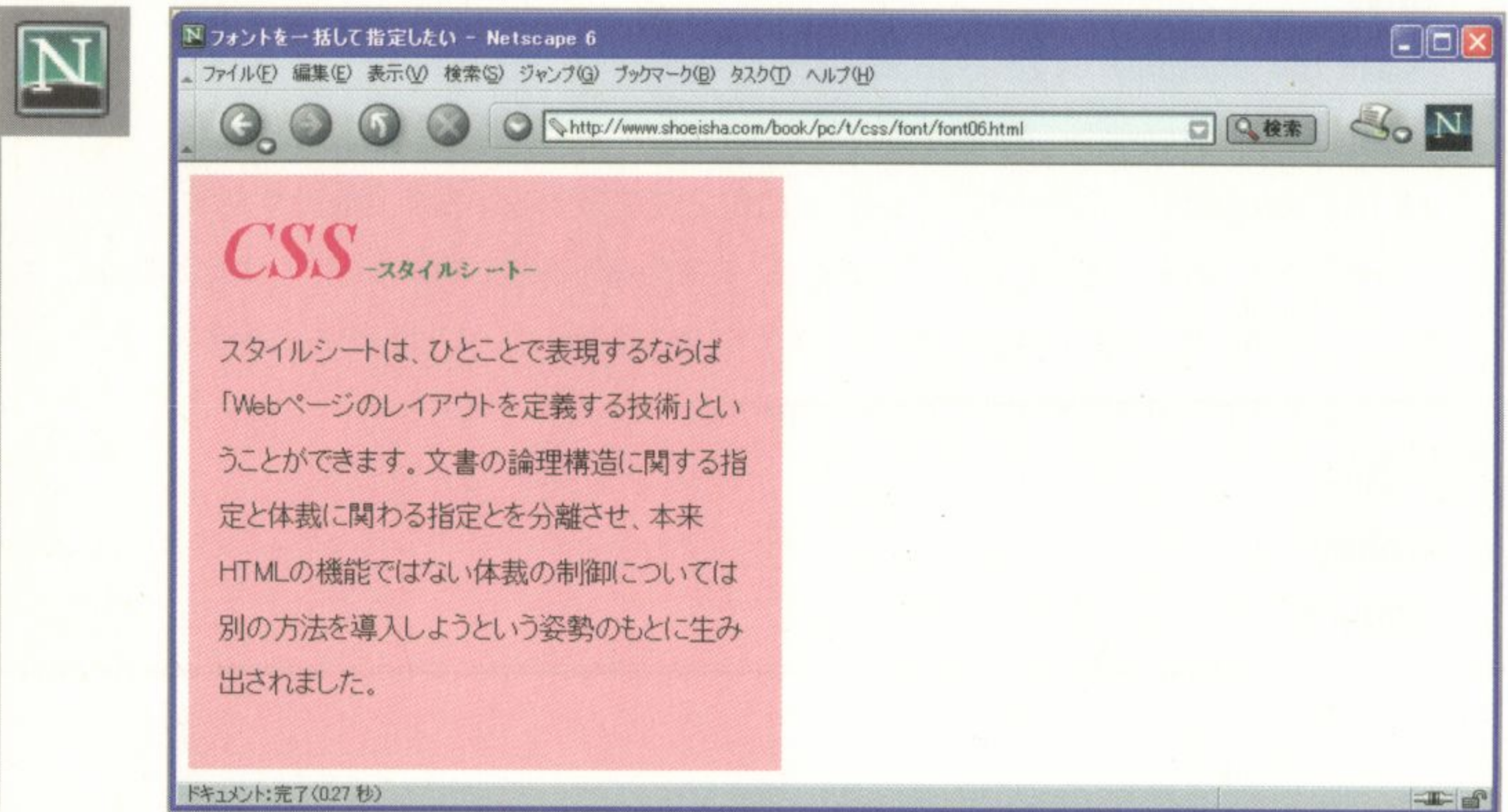
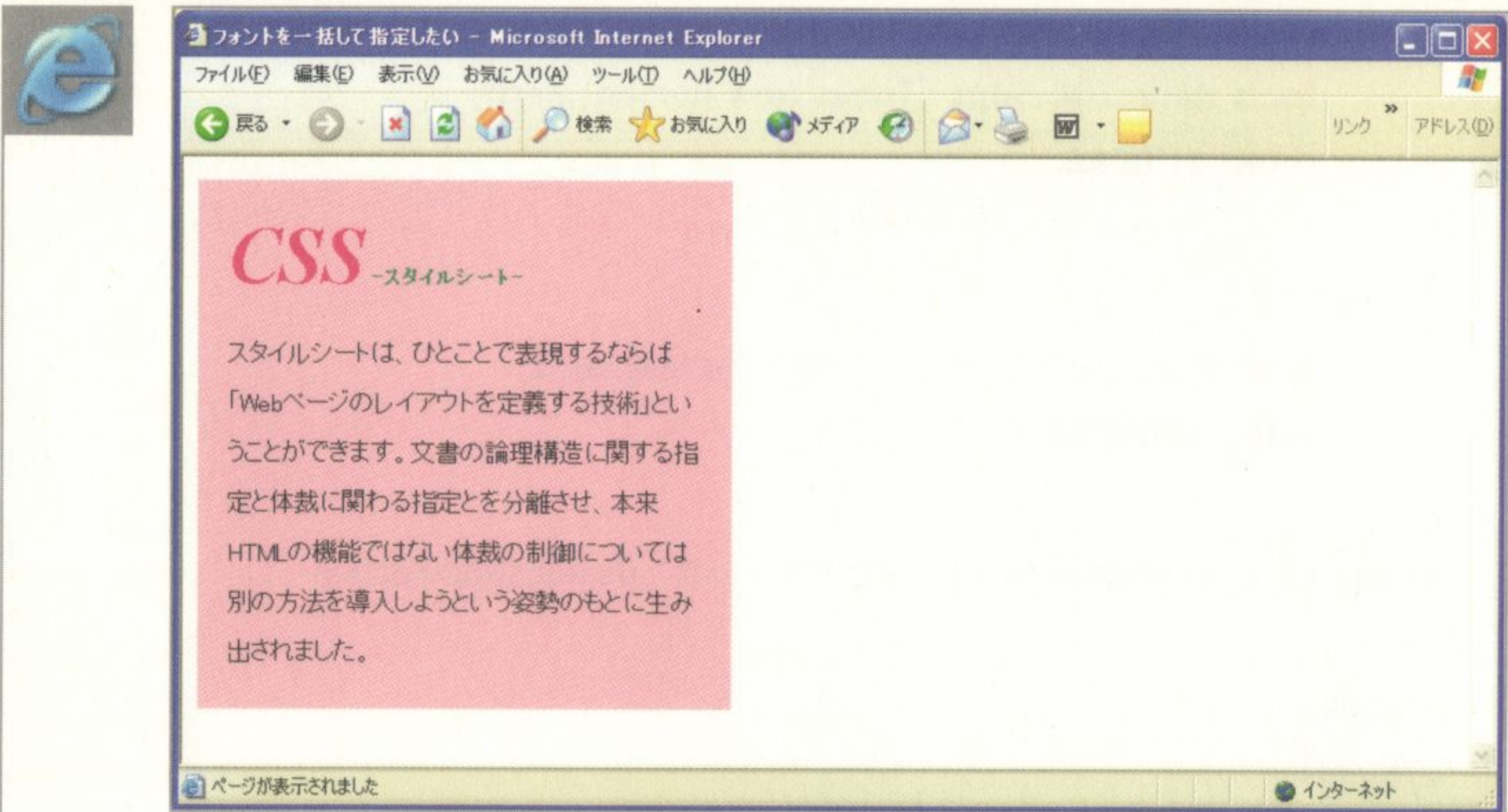
フォントに関する各種指定をまとめて設定します。

必要な値を半角スペースで区切って指定します。ただし、fontプロパティの場合は他の一括設定が可能なプロパティとは異なり、次のような規則があるので注意が必要です。

- font-style、font-variant、font-weightは、順序を入れ替えて指定することができる
- line-heightはfont-sizeのあとにスラッシュ（/）をつけて数値を指定
- font-sizeとfont-familyの値は省略できない
- 各属性の値を省略した場合は、それぞれのデフォルト値が適用される

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>フォントを一括して指定したい</title>
<style type="text/css">
<!--
div          {
        width: 350px;
        padding: 20px;
        background-color: #ffcccc
}
```

```
#sample1      {
        font: italic 700 50px "Times New Roman",serif;
        color: #ff0066
}
#sample2 {
        font: bold x-small "ＭＳ Ｐ明朝",平成明朝,serif;
        color: #008080
}
#sample3 { font: small/2em "ＭＳ Ｐゴシック",Osaka,sans-serif }
-->
</style>
</head>
<body>
<div>
<span id="sample1">CSS</span>
<span id="sample2">-スタイルシート-</span>
<p id="sample3">スタイルシートは、ひとことで表現するならば「Webページのレイア
ウトを定義する技術」ということができます。文書の論理構造に関する指定と体裁に関わる
指定とを分離させ、本来HTMLの機能ではない体裁の制御については別の方法を導入しよう
という姿勢のもとに生み出されました。</p>
</div>
</body>
</html>
```

フォント

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
共通	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照

行の高さを設定したい ……………………… p.66
フォントを指定したい ……………………… p.114
フォントサイズを指定したい ………………… p.117
フォントの太さを指定したい ………………… p.122
フォントを斜体にしたい …………………… p.126
文字にスモールキャピタルを指定したい ………… p.128
システムフォントを利用したい ……………… p.133

7 FONT

フォント

システムフォントを利用したい

font: ★

★……キーワード

システムフォントを指定します。

Windowsをはじめとする GUI 環境では、メニューやメッセージボックスなどで使用するフォントの設定があらかじめ用意されており、スタイルシートからその設定を呼び出して、Webページに使うことができます。

値には以下のキーワードがあります。

icon	アイコンの名前に使用されるフォント
menu	メニューで使用されるフォント
message-box	ダイアログボックスで使用されるフォント
caption	キャプションやラベルに使用されるフォント
status-bar	ウインドウのステータスバーで使用されるフォント

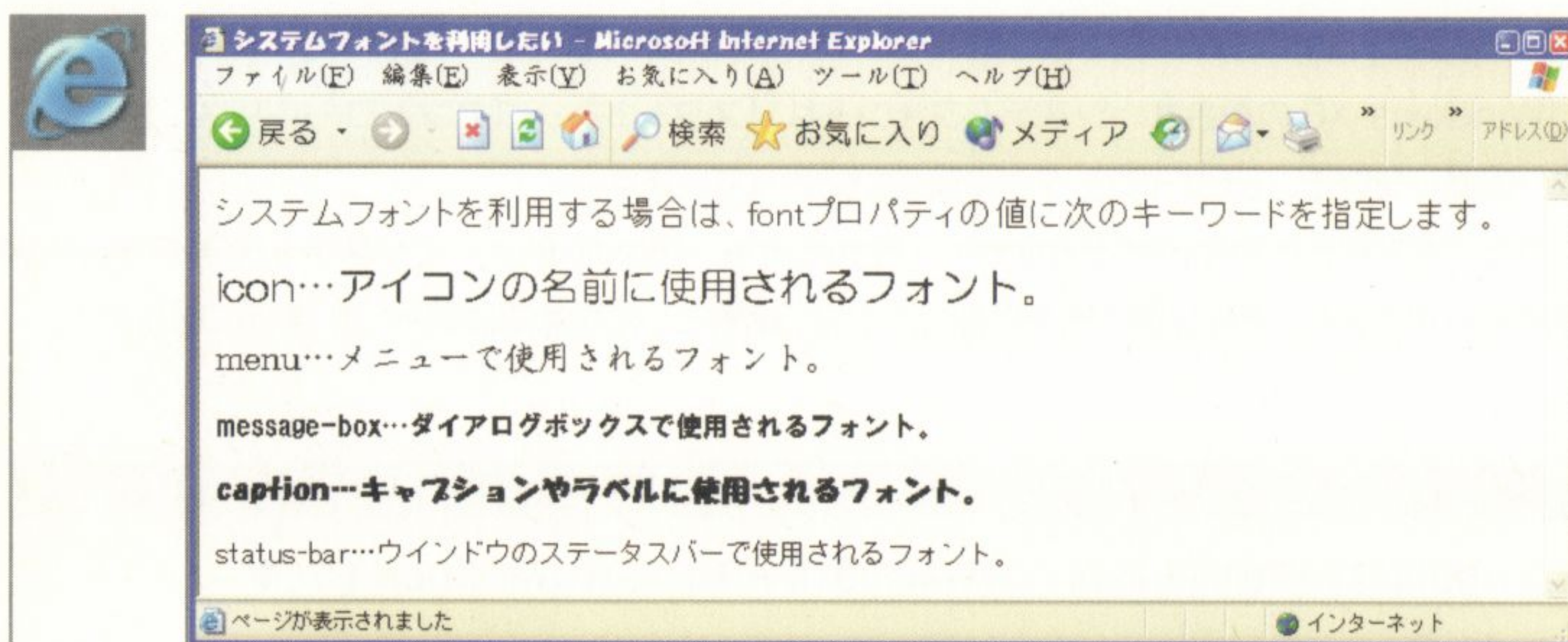
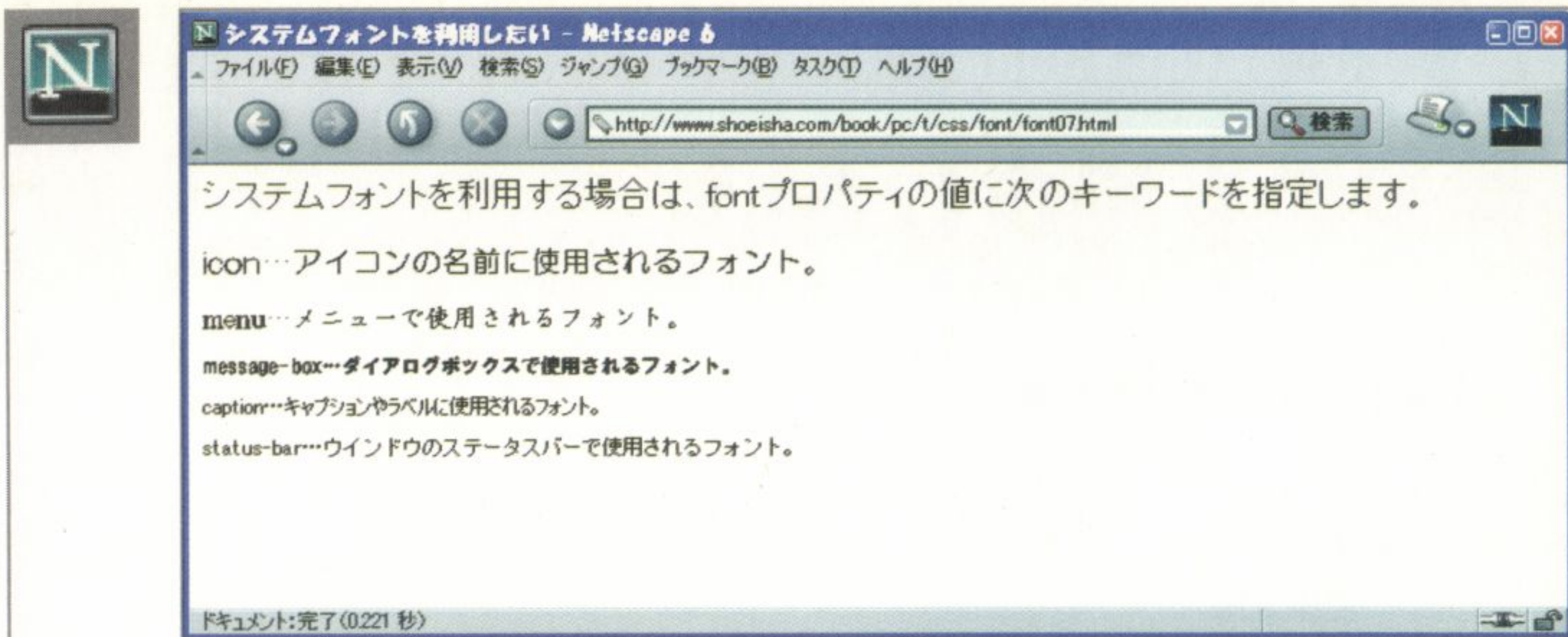
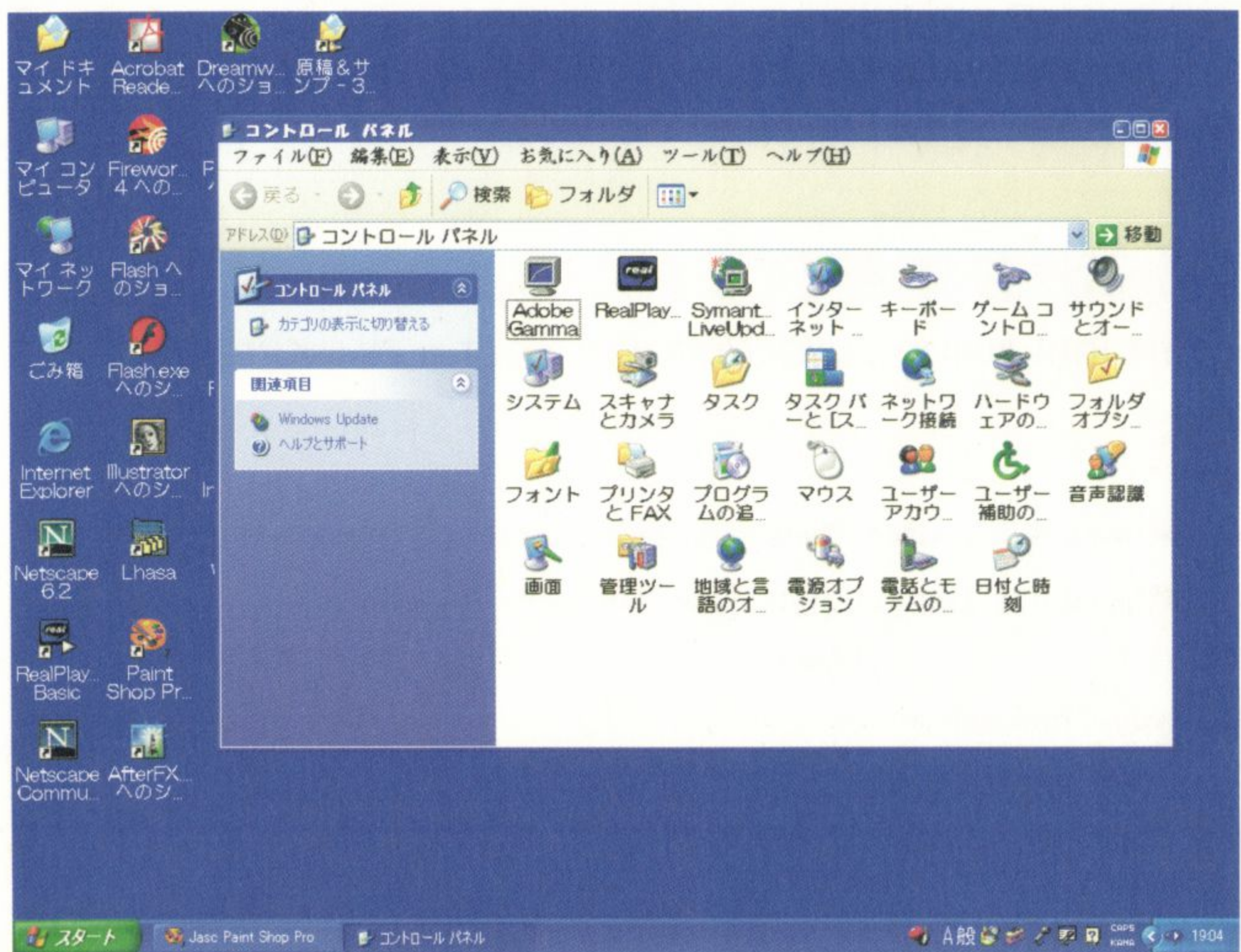
Windows XPの場合は、システムフォントは［スタート］→［コントロールパネル］を選んで［画面］を開き、［デザイン］タブに設定が記されています。

なお、本来ならばInternet ExplorerとNetscape（Navigator）でシステムの同一の値を参照するはずですが、実際は違う値を参照しており、結果として両者では異なった表示になります。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>システムフォントを利用したい</title>
<style type="text/css">
<!--
```

フォント

```
.sample1      { font: icon }
.sample2      { font: menu }
.sample3      { font: message-box }
.sample4      { font: caption }
.sample5      { font: status-bar }
-->
</style>
</head>
<body>
<p> システムフォントを利用する場合は、font プロパティの値に次のキーワードを指定します。</p>
<p class="sample1">icon …アイコンの名前に使用されるフォント。</p>
<p class="sample2">menu …メニューで使用されるフォント。</p>
<p class="sample3">message-box …ダイアログボックスで使用されるフォント。</p>
<p class="sample4">caption …キャプションやラベルに使用されるフォント。</p>
<p class="sample5">status-bar …ウインドウのステータスバーで使用されるフォント。</p>
</body>
</html>
```

▲ユーザーのシステムの設定によって表示されるフォントは異なります

▲同じシステムであってもNetscapeとInternet Explorerでは一部の表示フォントが異なります

▲ブラウザの画面は、システムにこのようなフォント設定をしている場合の表示例となります

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
icon	×	○	○	○	×	×	○
menu	×	○	○	○	×	×	○
message-box	×	×	○	○	×	×	○
status-bar	×	×	○	○	×	×	○
caption	×	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

フォントを一括して指定したい…………………p.130

1 BACKGROUND

背景色を指定したい

background-color: ★

★••••••キーワード
　　　　色指定値

要素の背景色を指定します。

色はRGB値、キーワード、transparent（透明）のいずれかで指定します。transparentを指定すると背景が透明になり、結果として親ボックスの背景色が透けて見えるようになります。

色の詳しい指定方法についてはp.47を参照してください。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>背景色を指定したい</title>
<style type="text/css">
<!--
body  { background-color: #ffcc99 }
h1    {
      background-color: #ff0000;
      color: #ffffff
 }
div   {
      width: 600px;
      padding: 20px;
      font-size: 12pt;
      line-height: 2em;
      background-color: #ffffcc
```

```
}
span    {
        border: 2px dotted #ff0066;
        background-color: #33ffcc
}
-->
</style>
</head>

<body>
<h1>スタイルシートとは</h1>
<div>
Webページを記述するHTMLは、<span>文書の論理的な構造を示す言語</span>です。
たとえば、見出しがあり、本文の中に段落やリストがあり……といったように、文書がどの
ような要素で構成されているのか、またある特定の部分が文書全体の中でどのような意味を
もっているのかを、コンピュータに知らせるための言語なのです。表現方法を指定するため
の言語ではありません。しかし実際は、Webの発展にともない、色やフォントサイズの指
定、レイアウトのためのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するよ
うになっていきました。W3Cではこの状況を改めるため、<span>構造に関する指定と体
裁に関わる指定とを分離させ</span>、HTMLの機能ではない体裁の制御については別の
方法を導入しようと考えるようになりました。こうした姿勢のもとに生み出されたのが
<span>スタイルシートの概念</span>です。
</div>
</body>

</html>
```

背景色を指定したい - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る 検索 お気に入り メディア
リンク アドレス(D)
スタイルシートとは
Webページを記述するHTMLは、文書の論理的な構造を示す言語です。たとえば、見出しがあり、本文の中に段落やリストがあり……といったように、文書がどのような要素で構成されているのか、またある特定の部分が文書全体の中でどのような意味をもっているのかを、コンピュータに知らせるための言語なのです。表現方法を指定するための言語ではありません。しかし実際は、Webの発展にともない、色やフォントサイズの指定、レイアウトのためのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するようになっていきました。W3Cではこの状況を改めるため、構造に関する指定と体裁に関わる指定とを分離させ、HTMLの機能ではない体裁の制御については別の方法を導入しようと考えるようになりました。こうした姿勢のもとに生み出されたのがスタイルシートの概念です。
ページが表示されました
インターネット

背景色を指定したい - Netscape 6
ファイル(F) 編集(E) 表示(V) 検索(S) ジャンプ(G) ブックマーク(B) タスク(T) ヘルプ(H)
http://www.shoeisha.com/book/pc/t/css/back/back01.html
検索
スタイルシートとは
Webページを記述するHTMLは、文書の論理的な構造を示す言語です。たとえば、見出しがあり、本文の中に段落やリストがあり……といったように、文書がどのような要素で構成されているのか、またある特定の部分が文書全体の中でどのような意味をもっているのかを、コンピュータに知らせるための言語なのです。表現方法を指定するための言語ではありません。しかし実際は、Webの発展にともない、色やフォントサイズの指定、レイアウトのためのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するようになっていきました。W3Cではこの状況を改めるため、構造に関する指定と体裁に関わる指定とを分離させ、HTMLの機能ではない体裁の制御については別の方法を導入しようと考えるようになりました。こうした姿勢のもとに生み出されたのがスタイルシートの概念です。
ドキュメント:完了(0.22 秒)

CSSにおける背景

CSSで「背景」という場合、要素が生成するボックスのうち、内容領域とパディングの背景を意味します。ボーダーの色は専用のプロパティ（border-colorなど）で設定し、マージンは常に透明なので親ボックスの背景が透けて見えることになります。ボックスの概念についてはp.45を参照してください。

背景色を指定するHTMLタグをCSSに改める

HTMLタグで背景色を指定するには、次のように背景色を指定できるタグのbgcolor属性を利用します。

<body bgcolor="★">～</body>	ページ全体の背景色
<table bgcolor="★">～</table>	テーブルの背景色
<◇ bgcolor="★">～</◇>	セルの背景色（◇──tr、th、td）
<marquee bgcolor="★">～</marquee>	マーキーの背景色

★──色の指定値

HTMLタグのbgcolor属性はDeprecated（推奨しない）とされており、背景色はスタイルシートで指定することが推奨されています。また、スタイルシートを使えば上記のタグに限らずすべての要素に背景色を指定することができます。同様の効果をスタイルシートで表現すると以下のようになります。

body	{ background-color: ★ }	ページ全体の背景色
table	{ background-color: ★ }	テーブルの背景色
◇	{ background-color: ★ }	セルの背景色（◇──tr、th、td）
marquee	{ background-color: ★ }	マーキーの背景色

★──色の指定値

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
色名	○	○	○	○	○	○	○
システムカラー	○	○	○	○	×	×	○
transparent	○	○	○	○	×	×	○
#rgb	○	○	○	○	○	○	○
#rrggbb	○	○	○	○	○	○	○
rgb(%,%,%)	○	○	○	○	○	○	○
rgb(,,)	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

文字色を指定したい……………………………p.56
背景画像を指定したい…………………………p.140
背景を一括して指定したい……………………p.157

2 BACKGROUND

背景画像を指定したい

background-image: ★

★••••••URL
　　　キーワード

要素の背景に画像を配置します。
値には次のような指定方法があります。

URL

background-image: url("☆");　　　☆……画像ファイルのURL

表示させる画像ファイルのURLを上記の形式で指定します。
なお、HTML文書から外部スタイルシートを読み込む場合は、HTML文書からの相対URLではなく、スタイルファイルからの相対URLで指定しなくてはなりません。詳細はp.50を参照してください。

キーワード

none　　　画像を表示しない（デフォルト）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>背景画像を指定したい</title>
<style type="text/css">
<!--
body {
    background-image: url("cat1.gif");
    background-color: #ffcc99
}
```


背景

```
h1      {
        background-color: #ff0000;
        color: #ffffff
}
div     {
        width: 600px;
        padding: 20px;
        font-size: 12pt;
        line-height: 2em;
        background-image: url("bg1.gif");
        background-color: #ffffcc
}
span    {
        border: 2px dotted #ff0066;
        background-color: #33ffcc
}
-->
</style>
</head>
<body>
<h1>スタイルシートとは</h1>
<div>
Webページを記述するHTMLは、<span>文書の論理的な構造を示す言語</span>です。
たとえば、見出しがあり、本文の中に段落やリストがあり……といったように、文書がどの
ような要素で構成されているのか、またある特定の部分が文書全体の中でどのような意味を
もっているのかを、コンピュータに知らせるための言語なのです。表現方法を指定するため
の言語ではありません。しかし実際は、Webの発展にともない、色やフォントサイズの指
定、レイアウトのためのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するよ
うになっていきました。W3Cではこの状況を改めるため、<span>構造に関する指定と体
裁に関わる指定とを分離させ</span>、HTMLの機能ではない体裁の制御については別の
方法を導入しようと考えるようになりました。こうした姿勢のもとに生み出されたのが
<span>スタイルシートの概念</span>です。
</div>
</body>
</html>
```

背景

▲body 要素の背景に使用した cat1.gif

▲div 要素の背景に使用した bg1.gif

文字の読みやすい背景

背景に画像を使う場合、画像に合わせて文字の色も変更することがあります（p.56参照）。しかし、背景画像の表示に時間がかかったり、ユーザーが画像を表示しないよう設定しているときなどに、指定した色によっては文字が読めないこともあります。こうした状況を考えて、同時にbackground-colorプロパティで文字が読みやすい背景色を設定しておいたほうがよいでしょう。

背景画像を指定するHTMLタグをCSSに改める

HTMLタグで背景画像の貼り込みを指定するには、次のように背景画像を指定できるタグのbackground属性を利用します。

<body background="★">～</body>	ページ全体の背景画像
<table background="★">～</table>	テーブルの背景画像
<◇ background="★">～</◇>	セルの背景画像（◇──tr、th、td）

★──画像ファイルのURL

<body>タグのbackground属性はDeprecated（廃止予定）とされており、またテーブル関係の背景画像はHTML4.01では定義されていません。背景画像の指定にはなるべくスタイルシートを使用するようにしましょう。また、スタイルシートを使えば上記のタグに限らずすべての要素に背景画像を指定することができます。同様の効果をスタイルシートで表現すると以下のようになります。

body	{ background-image: ★ }	ページ全体の背景画像
table	{ background-image: ★ }	テーブルの背景画像
◇	{ background-image: ★ }	セルの背景画像（◇──tr、th、td）

★──画像ファイルのURL

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
url(URL)	○	○	○	○	○	○	○
none	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

背景画像の繰り返し方法を指定したい…………p.144
背景画像を固定したい……………………p.147
背景画像の位置を指定したい…………………p.150
背景を一括して指定したい……………………p.157

3 BACKGROUND

背景画像の繰り返し方法を指定したい

background-repeat: ★

★……キーワード

背景

背景画像の繰り返し方法を指定します。

背景画像を設定した場合、通常は画像がタイル状に敷き詰められて表示されますが、background-repeatプロパティでは背景画像を敷き詰めるかどうか、敷き詰める場合にはどのように敷き詰めるのかを指定できるようになります。

値には以下のキーワードがあります。

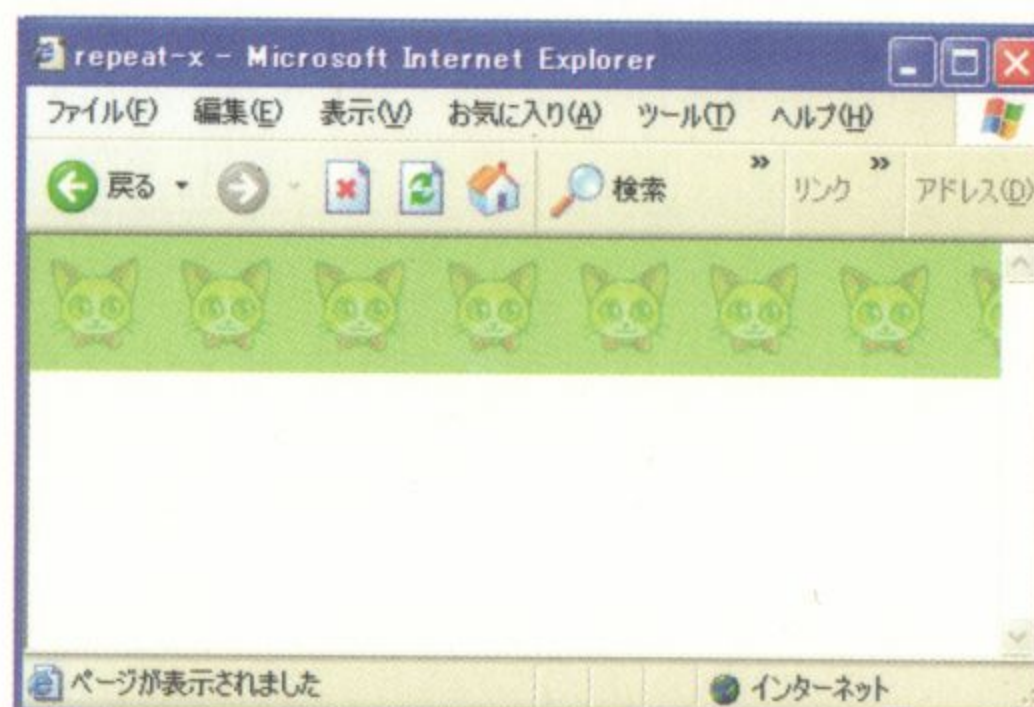
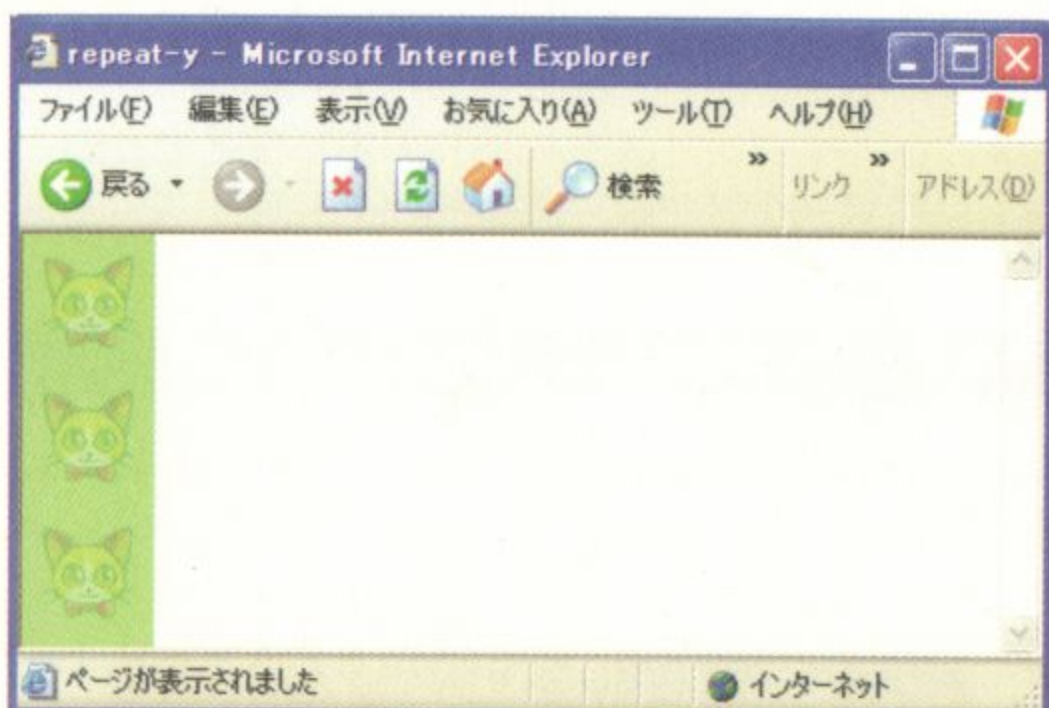
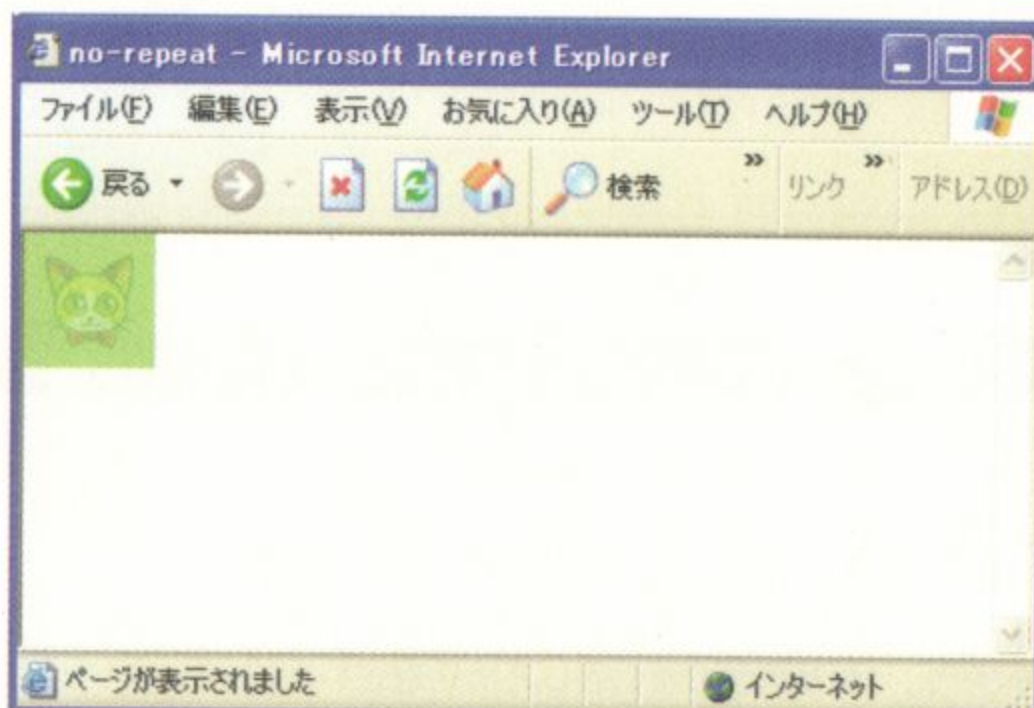
repeat	画面いっぱいに敷き詰めて表示（デフォルト）
repeat-x	横方向にのみ繰り返して表示
repeat-y	縦方向にのみ繰り返して表示
no-repeat	繰り返さずに、1点だけ表示

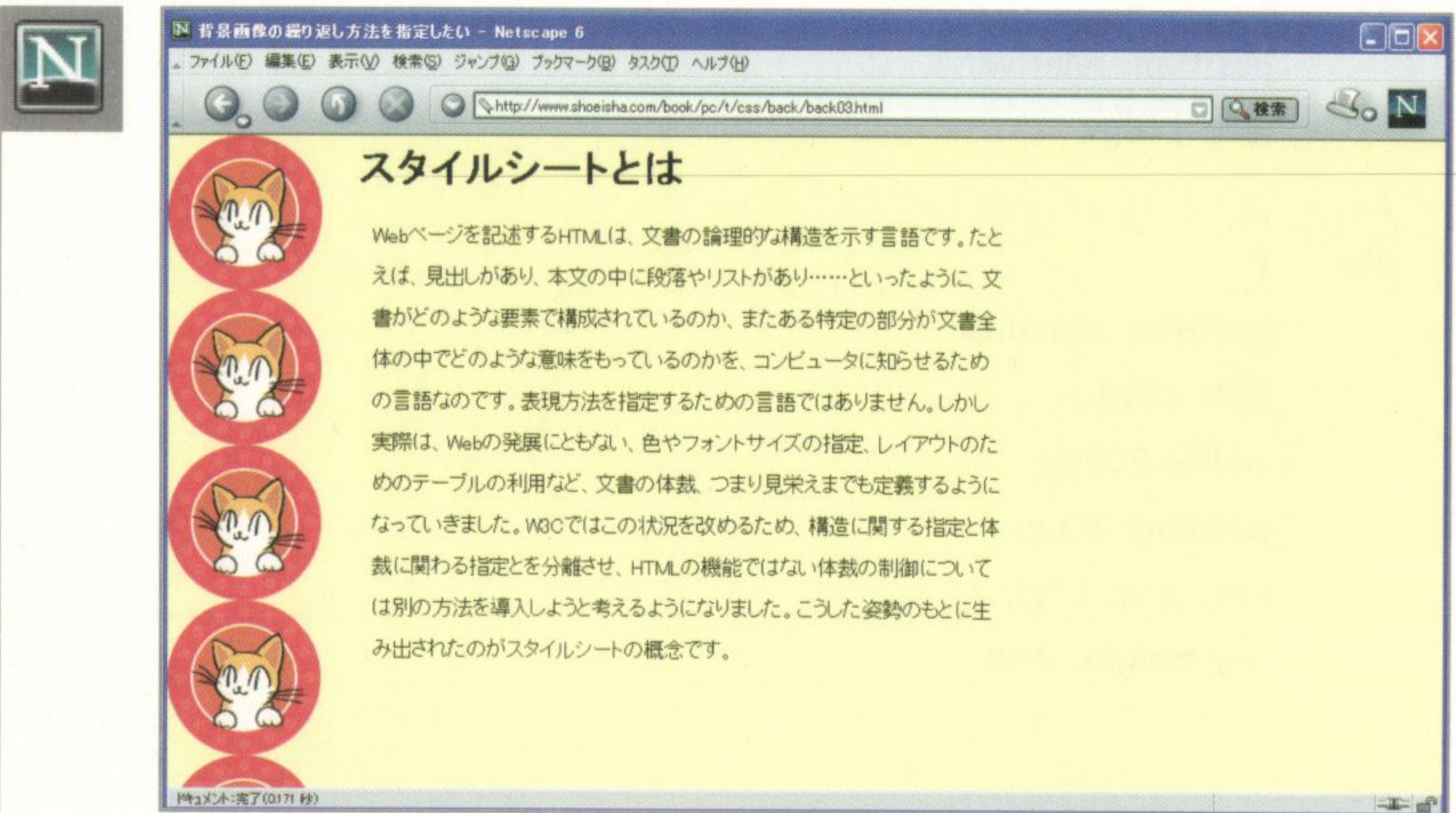
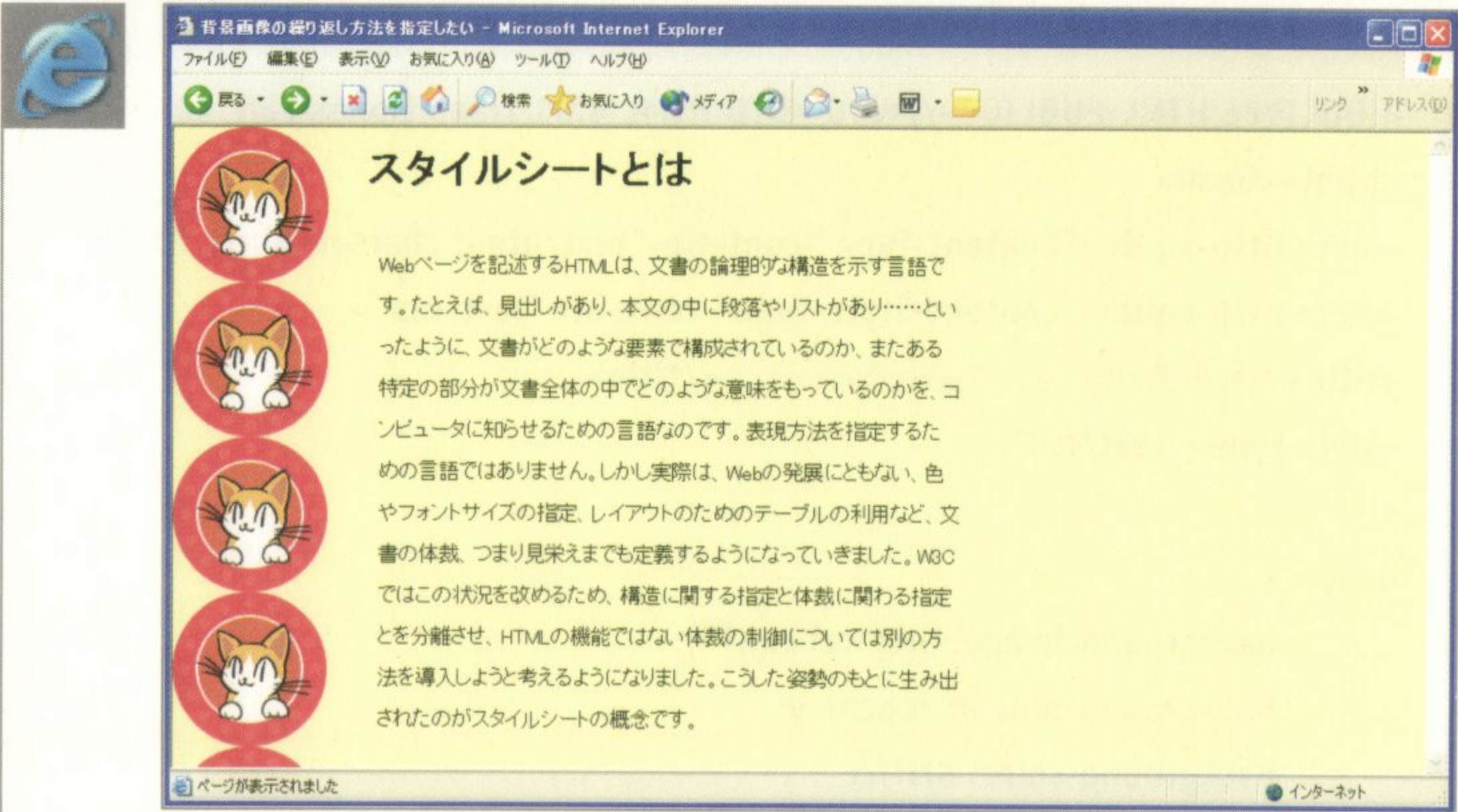
▲repeat

▲repeat-x

▲repeat-y

▲no-repeat

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>背景画像の繰り返し方法を指定したい</title>
<style type="text/css">
<!--
body	{
	background-image: url("cat2.gif");
	background-repeat: repeat-y;
	background-color: #ffffcc
}
h1	{
	position: relative;
	left: 140px
}
div	{
	position: absolute;
	left: 140px;
	width: 500px;
	padding: 20px;
	font-size: 12pt;
	line-height: 2em
}
-->
</style>
</head>
<body>
<h1>スタイルシートとは</h1>
<div>
Webページを記述するHTMLは、……（中略）……スタイルシートの概念です。
</div>
</body>
</html>
```

背景

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
repeat	○	○	○	○	○	○	○
repeat-x	○	○	○	○	○	○	○
repeat-y	○	○	○	○	○	○	○
no-repeat	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

背景画像を指定したい……………………………p.140
背景画像の位置を指定したい…………………………p.150
背景を一括して指定したい…………………………p.157

4 BACKGROUND

背景画像を固定したい

background-attachment: ★

★……キーワード

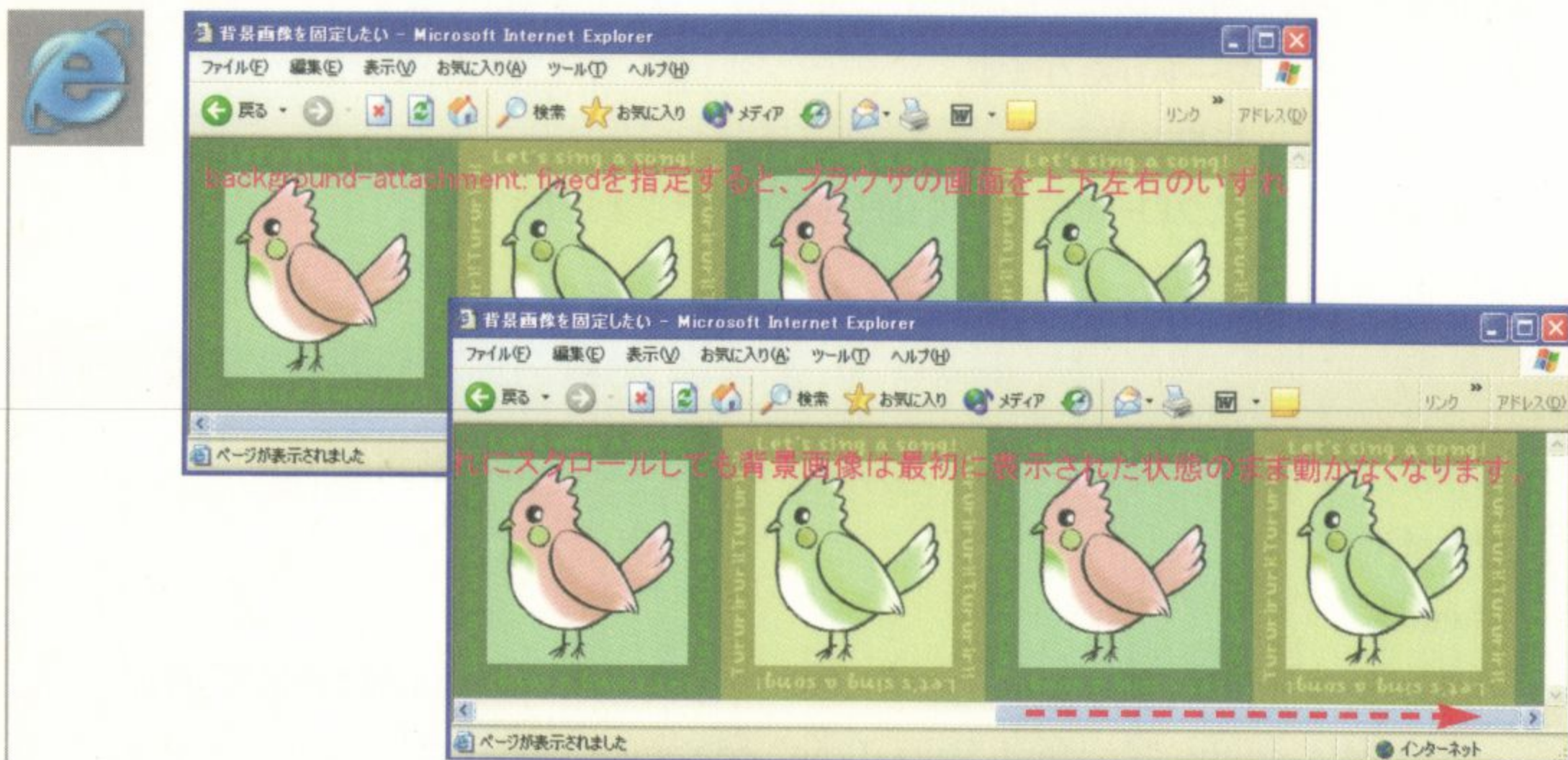
背景

背景画像をスクロールに応じて移動させるか、固定させるかを指定します。
値には以下のキーワードがあります。

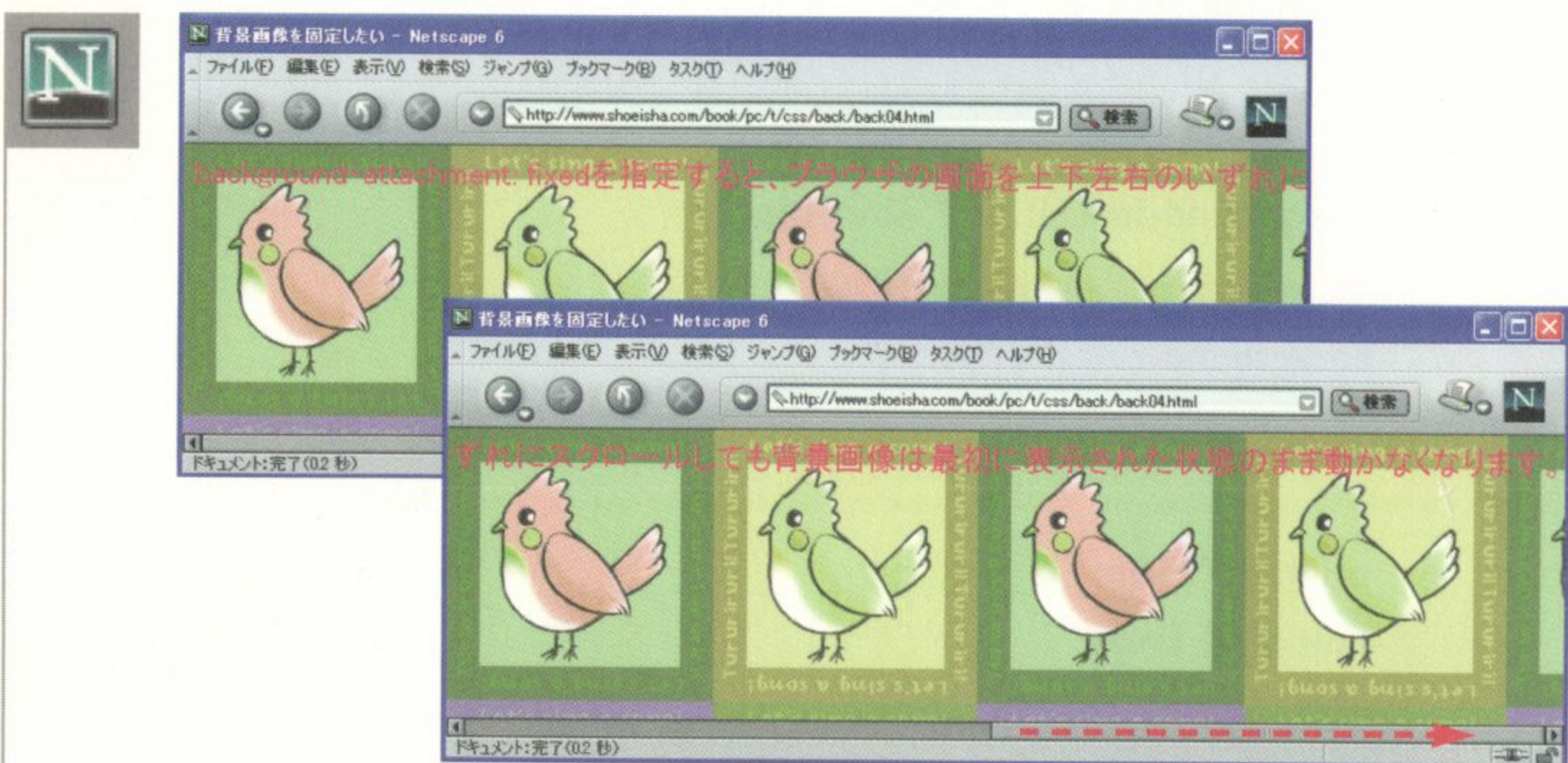
fixed	背景を固定
scroll	スクロールに応じて背景を移動（デフォルト）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>背景画像を固定したい</title>
<style type="text/css">
<!--
body	{
	background-image: url("4birds.jpg");
	background-attachment: fixed;
	background-color: #ffffff
}
p	{
	color: #ff3366;
	font-size: large;
	white-space: nowrap
}
-->
```

```
</style>
</head>
<body>
<p>
background-attachment: fixedを指定すると、ブラウザの画面を上下左右のいずれにスクロ
ールしても背景画像は最初に表示された状態のまま動かなくなります。
</p>
</body>
</html>
```

▲スクロールしても背景は動きません

▲スクロールしても背景は動きません

背景画像を固定するHTMLタグをCSSに改める

HTMLタグで背景画像の固定を指定するには、次のように<body background>のbgproperties属性でfixedを指定します。

<body background="★" bgproperties="fixed"> ～ </body>　　　　　　（★──画像ファイルのURL）

HTMLタグのbgproperties属性はInternet Explorerの独自拡張のためNetscape（Navigator）では無視されますが、スタイルシートでの指定はNetscape 6でも有効になります。同様の効果をスタイルシートで表現すると以下のようになります。

body { background-image: ★; background-attachment: fixed }　　　　（★──画像ファイルのURL）

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
fixed	○	○	○	○	×	×	○
scroll	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

背景画像を指定したい……………………p.140
背景画像の位置を指定したい……………………p.150
背景を一括して指定したい……………………p.157

背景画像の位置を指定したい

background-position: ★

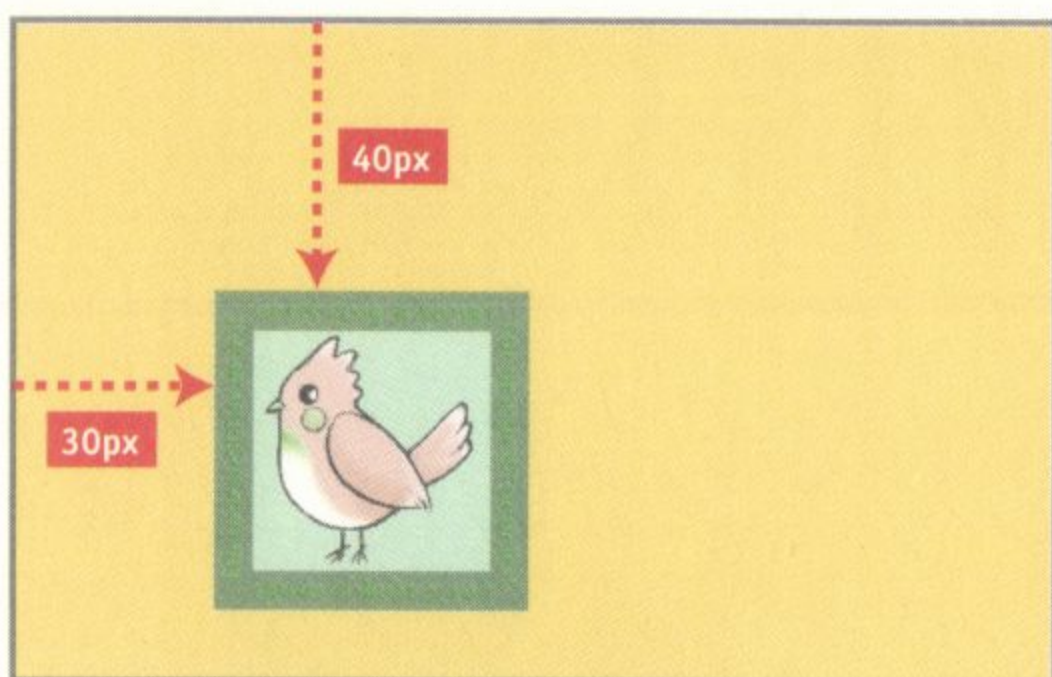
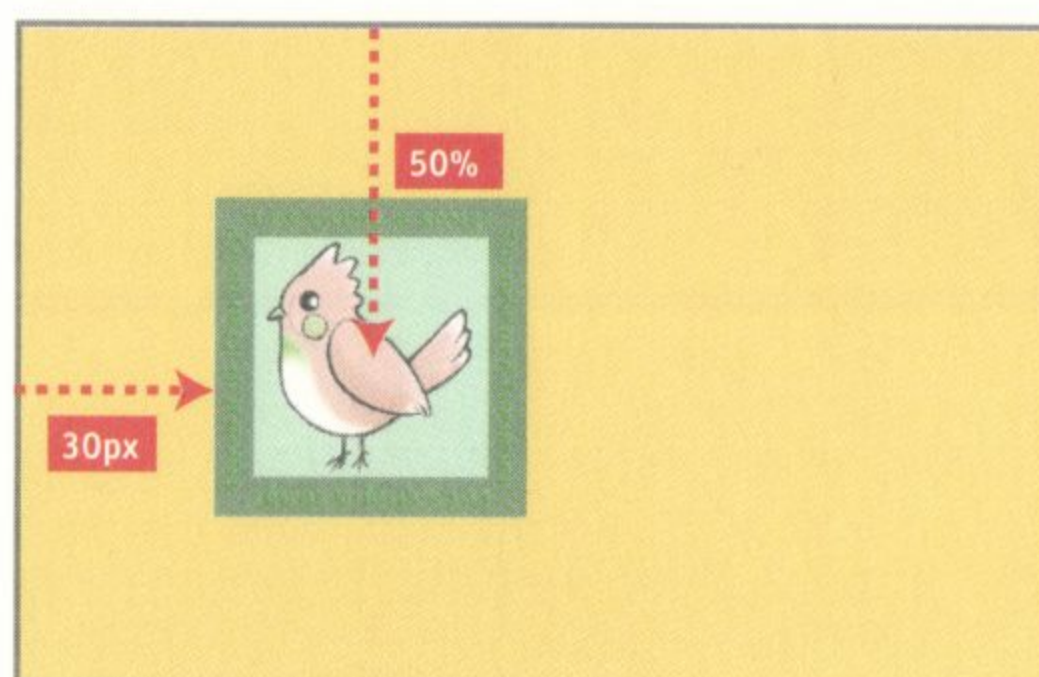
background-position: ★ ★

★……サイズを表す数値＋単位
パーセントを表す数値＋%
キーワード

背景画像の表示位置を指定します。

値には次のような指定方法があります

サイズを表す数値＋単位

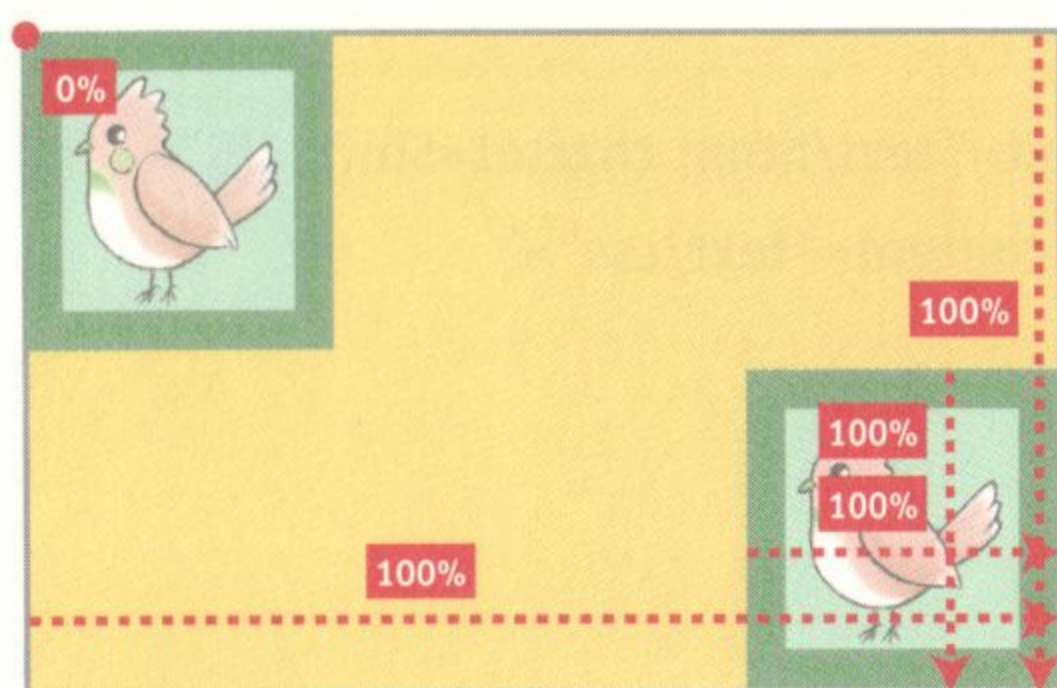
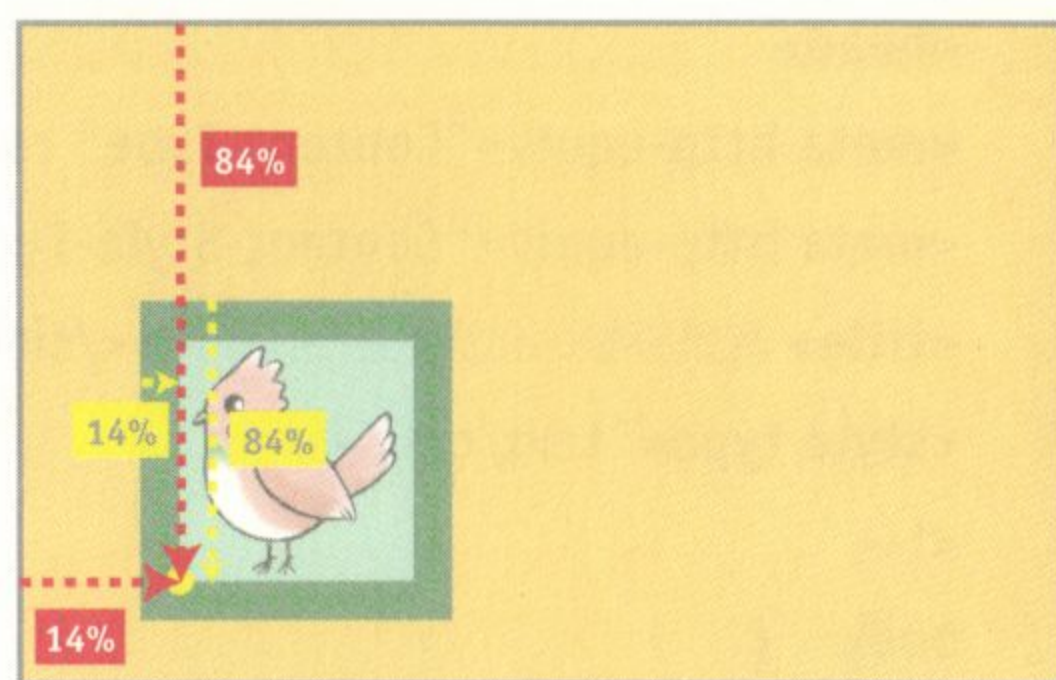
ボックス領域の左上を基点とし、水平方向の位置（左からの距離）と垂直方向の位置（上からの距離）を半角スペースで区切って指定します。たとえば、30px 40pxと指定した場合はパディングの左から30ピクセル、上から40ピクセルの位置に画像の左上を揃えることになります。値を1つだけ指定した場合は水平方向の位置を指定したことになり、垂直方向の位置は「50%」に設定されます。値には負の数値も指定することができます。

▲ background-position: 30px 40px

▲ background-position: 30px

パーセントを表す数値＋%

画像の位置をボックス領域の幅あるいは高さに対する割合で指定します。

ボックス領域の左上を基点とし、水平方向の位置（左からの距離）と垂直方向の位置（上からの距離）を半角スペースで区切って指定する点はサイズによる指定と同じですが、値の解釈が異なります。

パーセントで指定すると、ボックス領域の位置の割合と画像の位置の割合が揃います。たとえば、0% 0%と指定した場合はパディングの左上と画像の左上が揃い、14% 84%と指定した場合はパディングの左から14% 上から84%の位置と画像の左から14% 上から84%の位置が揃

います。同様に100% 100%と指定した場合にはパディングの右下と画像の右下が揃うことになります。値を1つだけ指定した場合は水平方向の位置を指定したことになり、垂直方向の位置は「50%」に設定されます。値には負の数値も指定することができます。

▲ background-position: 0% 0%
background-position: 100% 100%

▲ background-position: 14% 84%

キーワード

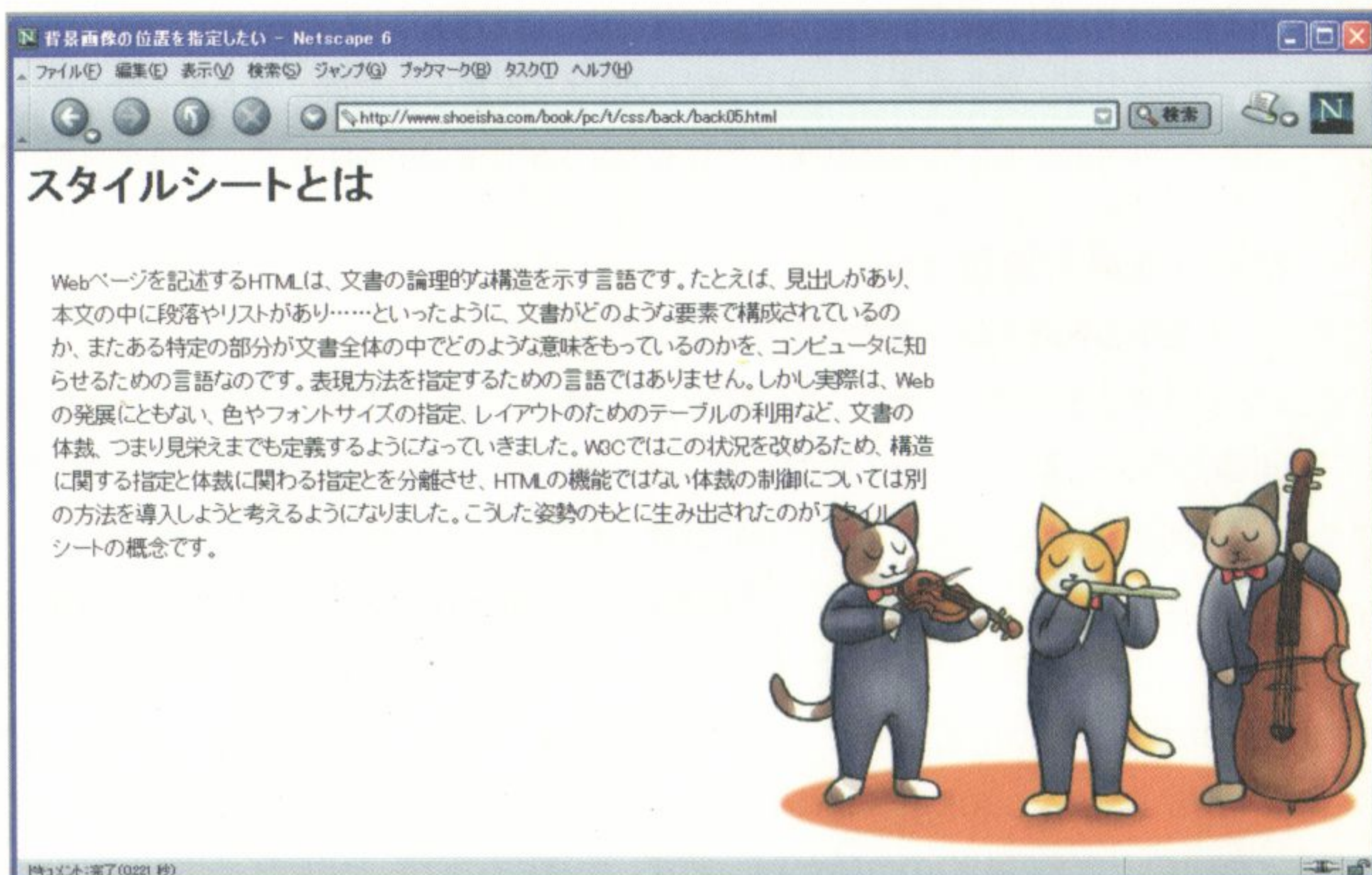
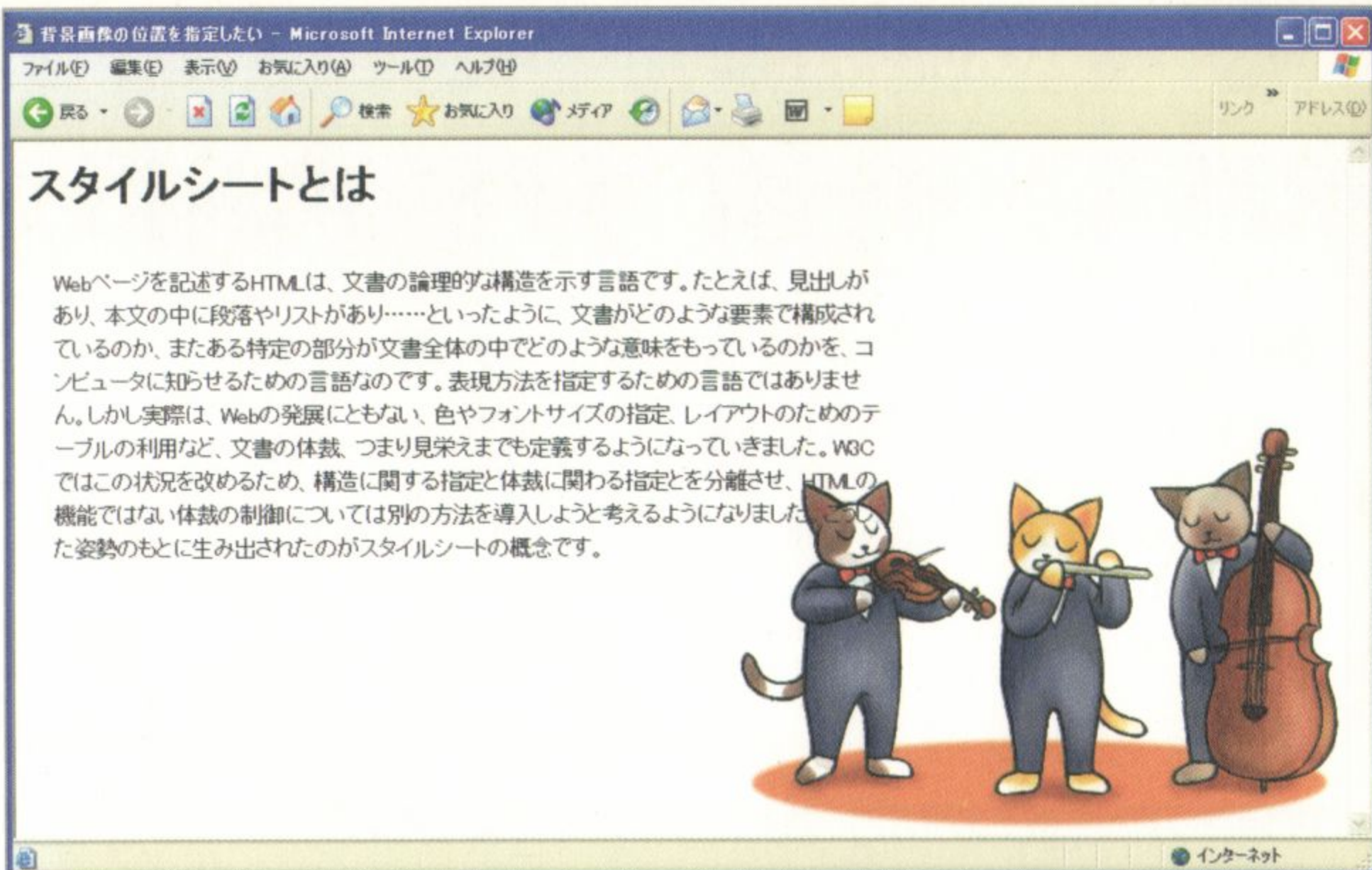
top	上端
center	中央
bottom	下端
left	左端
right	右端

垂直方向の位置を指定する場合はtop、center、bottom、水平方向を指定する場合はleft、center、rightを任意の順番で、半角スペースで区切って指定します。値を1つだけ設定した場合は、もう一方はcenterに設定されます。

数値による指定とパーセントによる指定とは組み合わせて使用することができます。しかしキーワードを数値による指定やパーセントによる指定と組み合わせて使用することはできませんので注意してください。

なお、背景画像を1点だけ指定した位置に表示させるには、同時にbackground-repeat属性でno-repeatを指定してください。それ以外の場合は、指定した位置を基点に繰り返し敷き詰められます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>背景画像の位置を指定したい</title>
<style type="text/css">
<!--
body    {
        background-image: url("catband.jpg");
        background-repeat: no-repeat;
        background-position: 100% 100%
}
div     {
        width: 650px;
        padding: 20px;
        font-size: 12pt;
        line-height: 1.5em;
}

-->
</style>
</head>
<body>
<h1>スタイルシートとは</h1>
<div>
Webページを記述するHTMLは、文書の論理的な構造を示す言語です。たとえば、見出しがあり、……（中略）……こうした姿勢のもとに生み出されたのがスタイルシートの概念です。
</div>
</body>
</html>
```

背景

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	×	×	○
パーセント	○	○	○	○	×	×	○
キーワード	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

参照
背景画像を指定したい……………………………p.140
背景画像の繰り返し方法を指定したい…………p.144
背景画像の位置を垂直・水平方向別に指定したい‥p.154
背景を一括して指定したい…………………………p.157

6 BACKGROUND

背景画像の位置を垂直・水平方向別に指定したい

background-position-x: ★ 水平方向
background-position-y: ★ 垂直方向

★……サイズを表す数値＋単位
パーセントを表す数値＋%
キーワード

背景

背景画像の表示位置を水平方向と垂直方向で個別に指定します。

値には次のような指定方法があります。

サイズを表す数値＋単位

ボックス領域の左上を基点とし、水平方向の位置（左からの距離）をbackground-position-xで、垂直方向の位置（上からの距離）をbackground-position-yで、それぞれ指定します。

パーセントを表す数値＋%

ボックス領域の幅あるいは高さに対する割合で指定します。

ボックス領域の左上を基点とし、水平方向の位置（左からの距離）をbackground-position-xで、垂直方向の位置（上からの距離）をbackground-position-yで、それぞれ指定する点はサイズによる指定と同じですが、値の解釈が異なります。

パーセントで指定すると、ボックス領域の幅あるいは高さの位置の割合と、画像の位置の割合が揃います。たとえば、0%と指定した場合には画像の左上とパディングの左上が揃い、background-position-x: 14%と指定した場合はパディングの左から14%と画像の左から14%の位置が揃い、background-position-y: 84%と指定した場合はパディングの上から84%と画像の上から84%の位置が揃います。同様に100%と指定した場合には、パディングの右端あるいは下端と画像の右端あるいは左端が揃うことになります。

キーワード

top	上端
center	中央
bottom	下端
left	左端
right	右端

水平方向の位置を指定する場合はleft、center、rightをbackground-position-xに対して指定します。

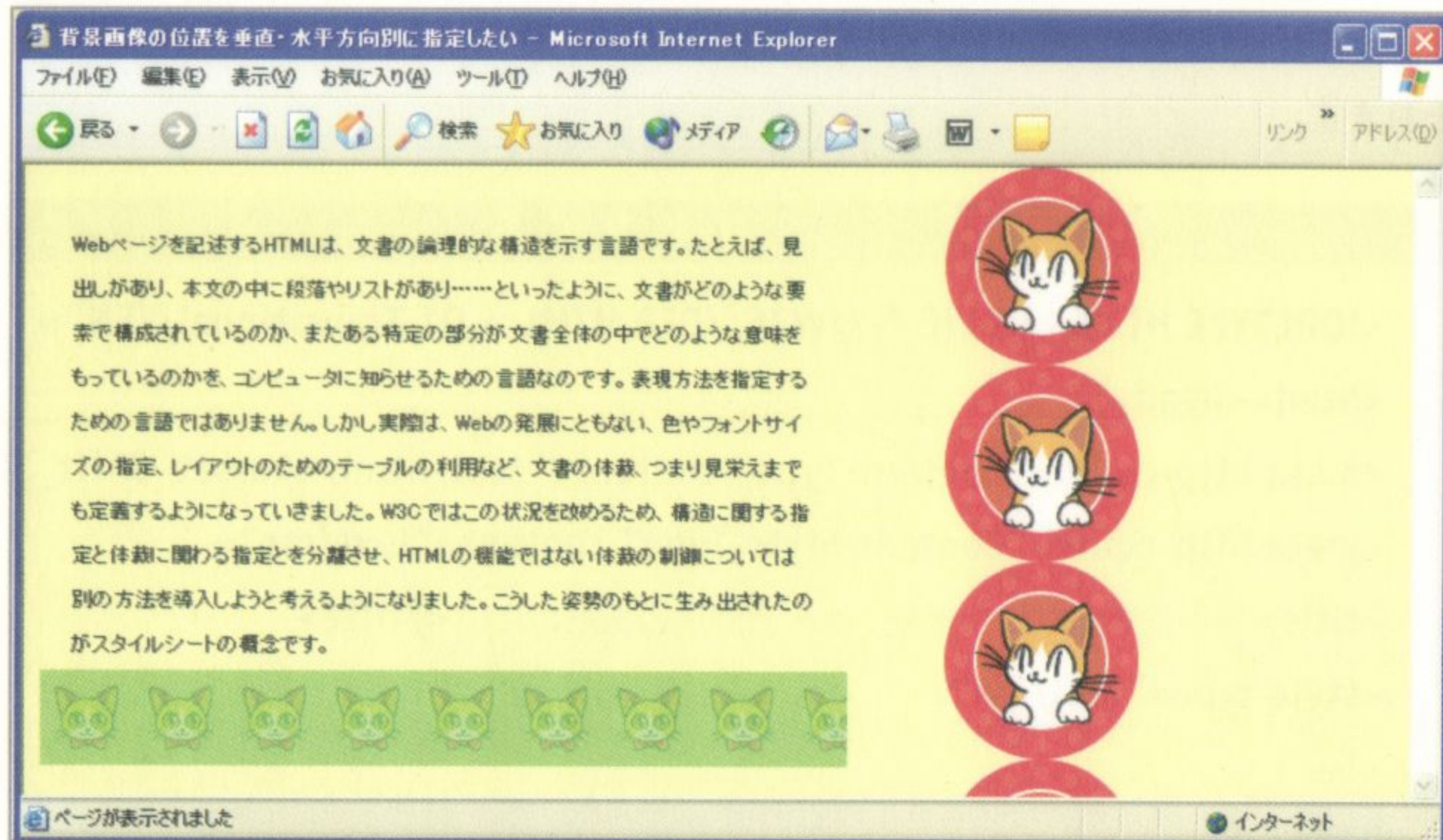
垂直方向の位置を指定する場合はtop、center、bottomをbackground-position-yに対して指定します。

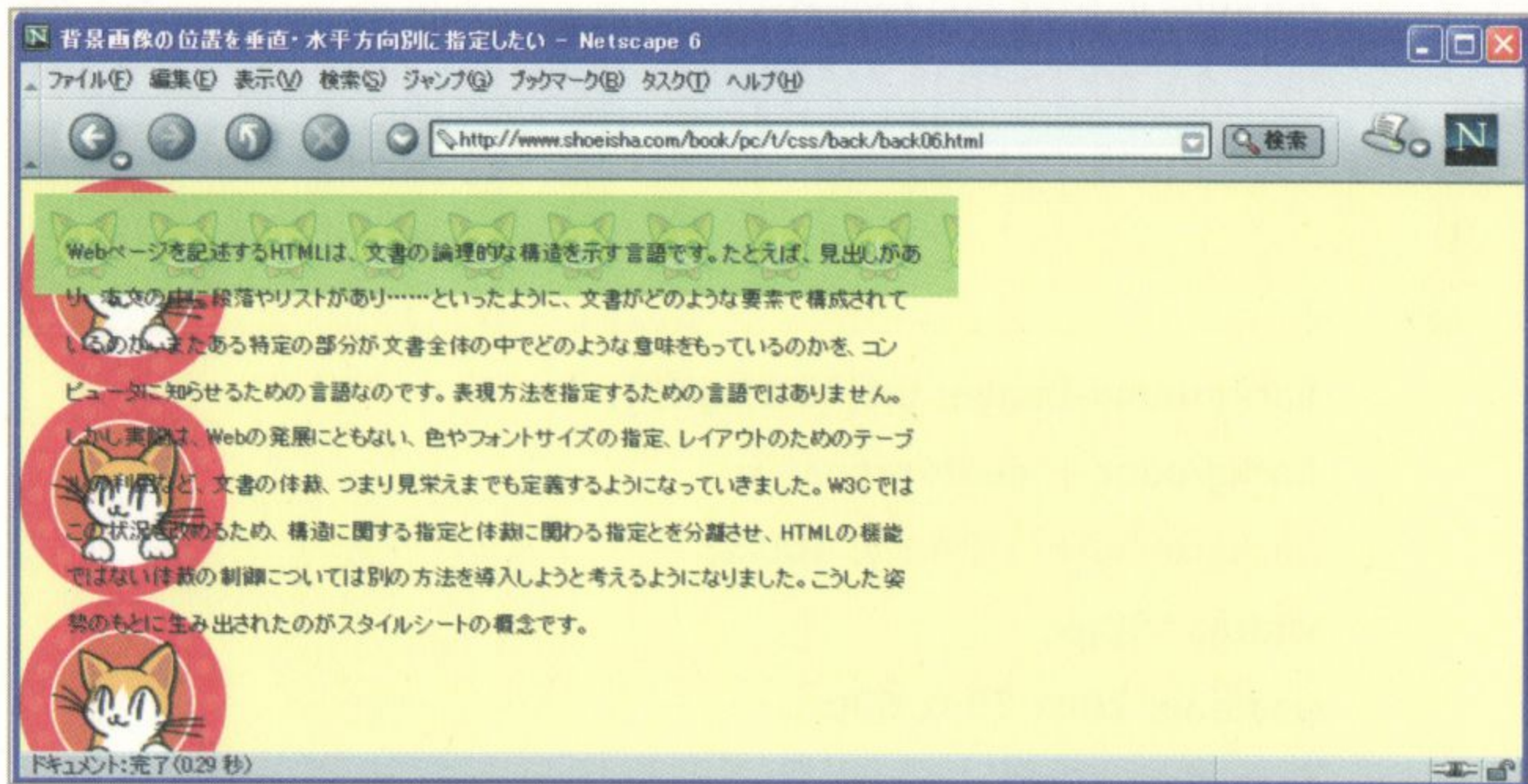
Internet Explorerが独自に拡張したプロパティのため、Netscapeでは対応していませんので注意してください。

なお、背景画像を1点だけ指定した位置に表示させるには、同時にbackground-repeat属性でno-repeatを指定してください。それ以外の場合は、指定した位置を基点に繰り返し敷き詰められます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>背景画像の位置を垂直・水平方向別に指定したい</title>
<style type="text/css">
<!--
body    {
        background-image: url("cat2.gif");
        background-repeat: repeat-y;
        background-position-x: 570px;
        background-color: #ffffcc
}
div     {
        background-image: url("cat1.gif");
        background-repeat: repeat-x;
        background-position-y: 100%;
        width: 500px;
        padding: 20px 20px 60px;
        font-size: 10pt;
        line-height: 2em
}
-->
</style>
</head>
<body>
```

```
<div>
Webページを記述するHTMLは、……（中略）……スタイルシートの概念です。
</div>
</body>
</html>
```


▲background-position属性よりも垂直・水平方向への指定を明確に行うことができます

▲Netscapeは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	×	×	×
パーセント	○	○	○	○	×	×	×
キーワード	○	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります

参照
背景画像を指定したい……………………………p.140
背景画像の繰り返し方法を指定したい…………p.144
背景画像の位置を指定したい……………………p.150

7 BACKGROUND

背景を一括して指定したい

background: ★ ☆ ▲ △ ■

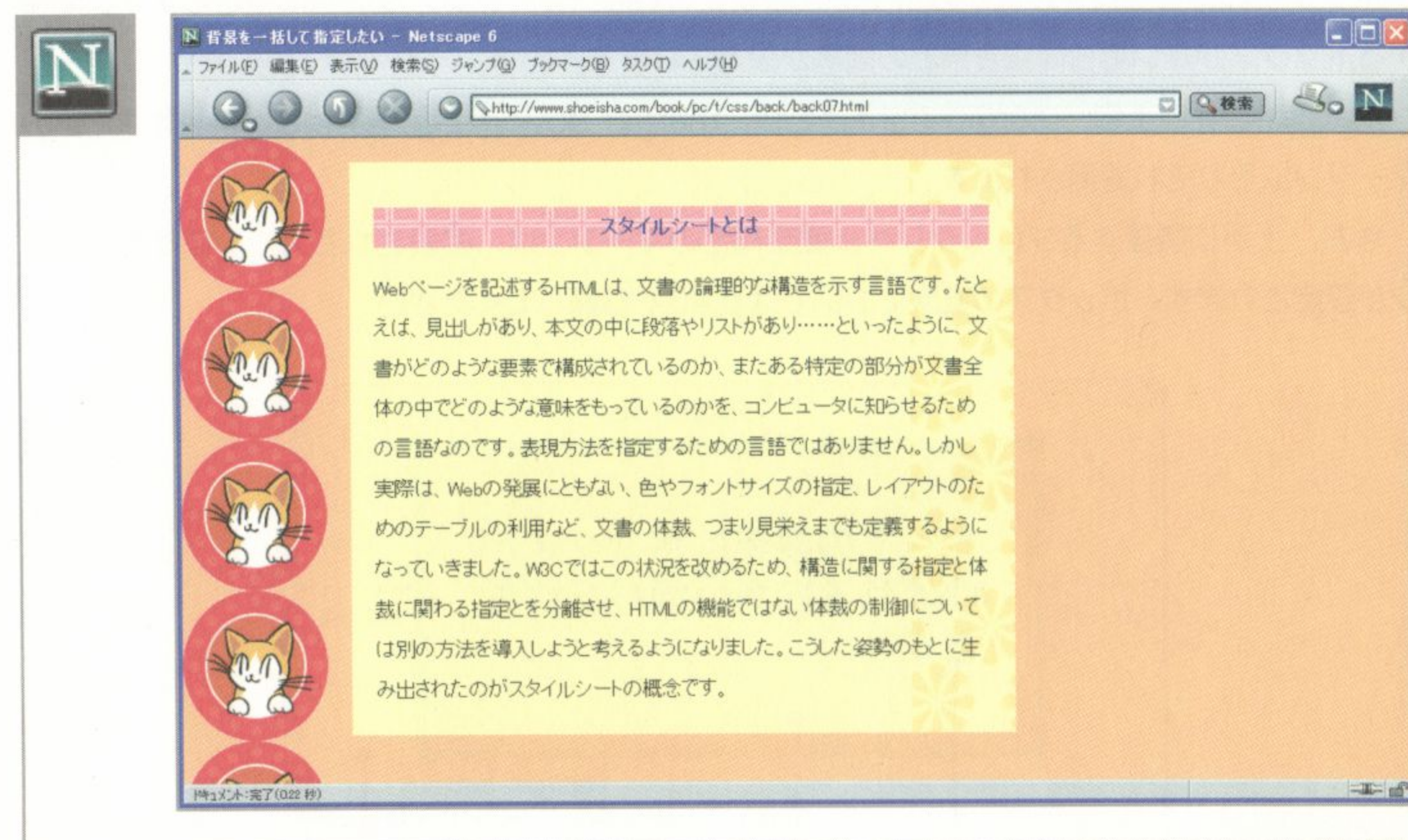
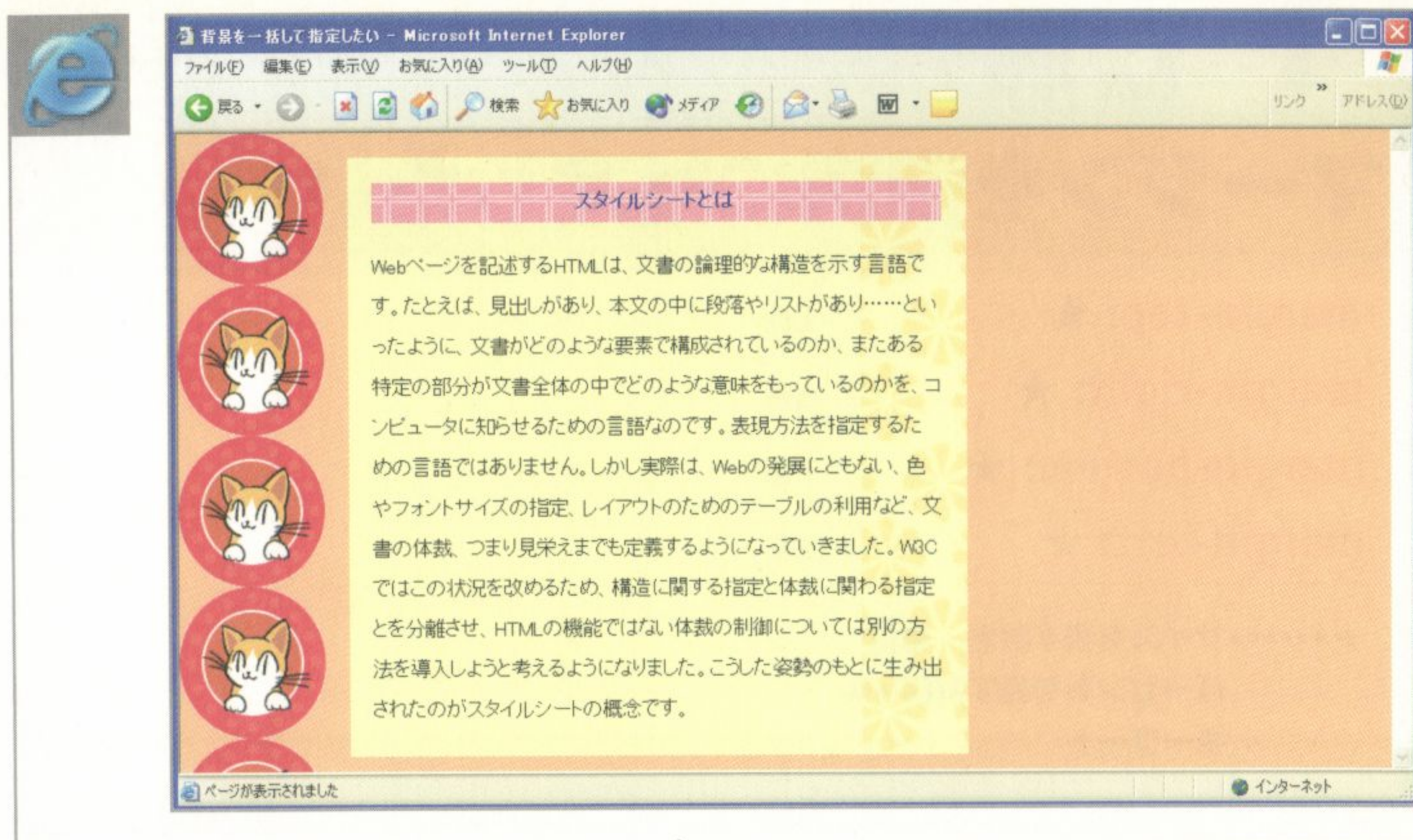
★••••••background-colorの値（背景色）
☆••••••background-imageの値（背景画像）
▲••••••background-repeatの値（背景画像の繰り返し）
△••••••background-attachmentの値（背景画像のスクロール）
□••••••background-positionの値（背景画像の位置）

背景に関する各種指定を一括して設定します。background-color（色）、background-image（画像）、background-repeat（繰り返し）、background-attachment（スクロール）、background-positionの値（位置）のうち必要な値を任意の順番で並べ、半角スペースで区切って指定します。省略された値についてはデフォルトの設定が適用されます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>背景を一括して指定したい</title>
<style type="text/css">
<!--
body    { background: #ffcc99 url("cat2.gif") fixed repeat-y }
div     {
        position: absolute;
        top: 20px;
        left:140px;
        width: 500px;
        padding: 20px;
        font-size: 12pt;
```

背景

```
        line-height: 2em;
        background: #ffffcc url("bg1.gif") repeat-y right
}
p       {
        text-align:center;
        font-weight: bold;
        color: navy;
        background: #ffcccc url("bg2.gif")
        }
-->
</style>
</head>

<body>
<div>
<p>スタイルシートとは</p>
Webページを記述するHTMLは、文書の論理的な構造を示す言語です。たとえば、見出しがあり、本文の中に段落やリストがあり……といったように、文書がどのような要素で構成されているのか、またある特定の部分が文書全体の中でどのような意味をもっているのかを、コンピュータに知らせるための言語なのです。表現方法を指定するための言語ではありません。しかし実際は、Webの発展にともない、色やフォントサイズの指定、レイアウトのためのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するようになっていきました。W3Cではこの状況を改めるため、構造に関する指定と体裁に関わる指定とを分離させ、HTMLの機能ではない体裁の制御については別の方法を導入しようと考えるようになりました。こうした姿勢のもとに生み出されたのがスタイルシートの概念です。
</div>
</body>

</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
共通	○	○	○	○	○*1	○*1	○

＊1：positon と attachment には対応していません

※適用するセレクタによっても効果が変わります

参照

背景色を指定したい……………………………p.136

背景画像を指定したい…………………………p.140

背景画像の繰り返し方法を指定したい…………p.144

背景画像を固定したい…………………………p.147

背景画像の位置を指定したい…………………p.150

1 BOX

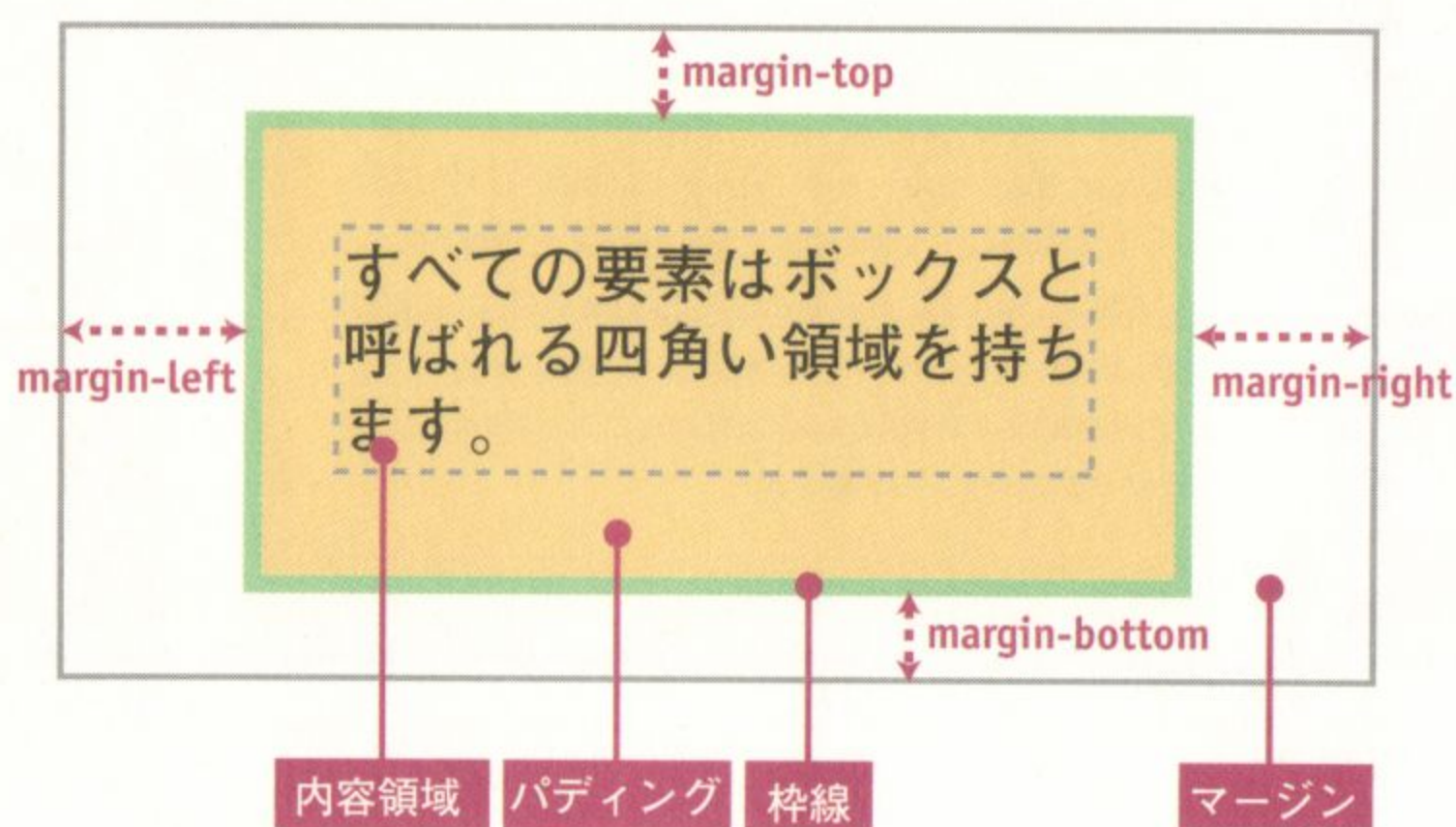
マージンを個別に指定したい

margin-top: ★ 上側
margin-right: ★ 右側
margin-bottom: ★ 下側
margin-left: ★ 左側

★……サイズを表す数値＋単位
　　パーセントを表す数値＋%
　　キーワード

ボックスの枠と隣接するほかのボックス領域との間隔（マージン）を個別に指定します。隣接するボックスがない場合は親要素との間隔が設定されます。

margin-topは上側、margin-rightは右側、margin-bottomは下側、margin-leftは左側のマージンにそれぞれ適用されます。

なお、上下に隣接するブロックレベル要素同士のマージンは、相殺されて大きいほうのマージンが設定されます。ボックスの概念についてはp.45を参照してください。

値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけてマージンを指定します。マイナスの数値を指定することもできます。単位についてはp.46を参照してください。

パーセントを表す数値＋%

親要素のボックス領域の幅に対する割合でマージンを指定します。左右だけでなく、上下のマージンについても幅を基準にします。

キーワード

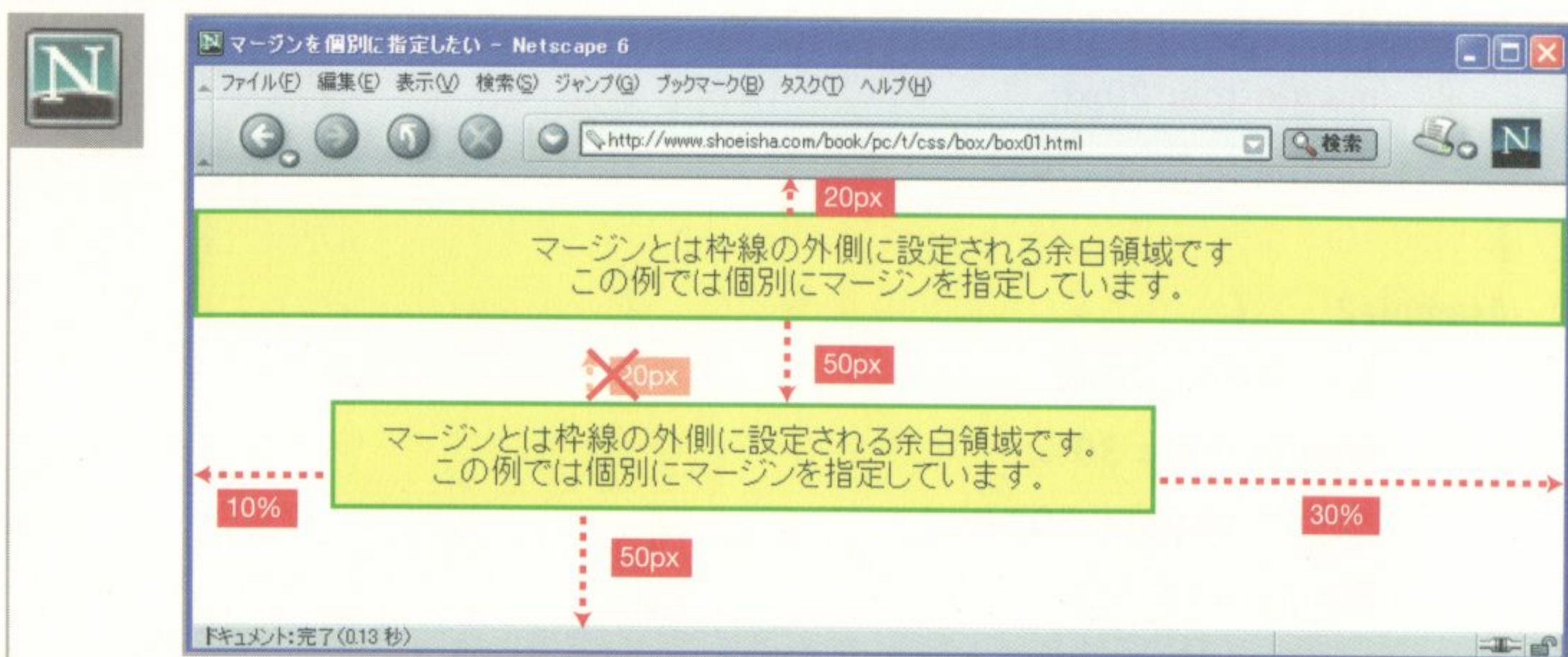
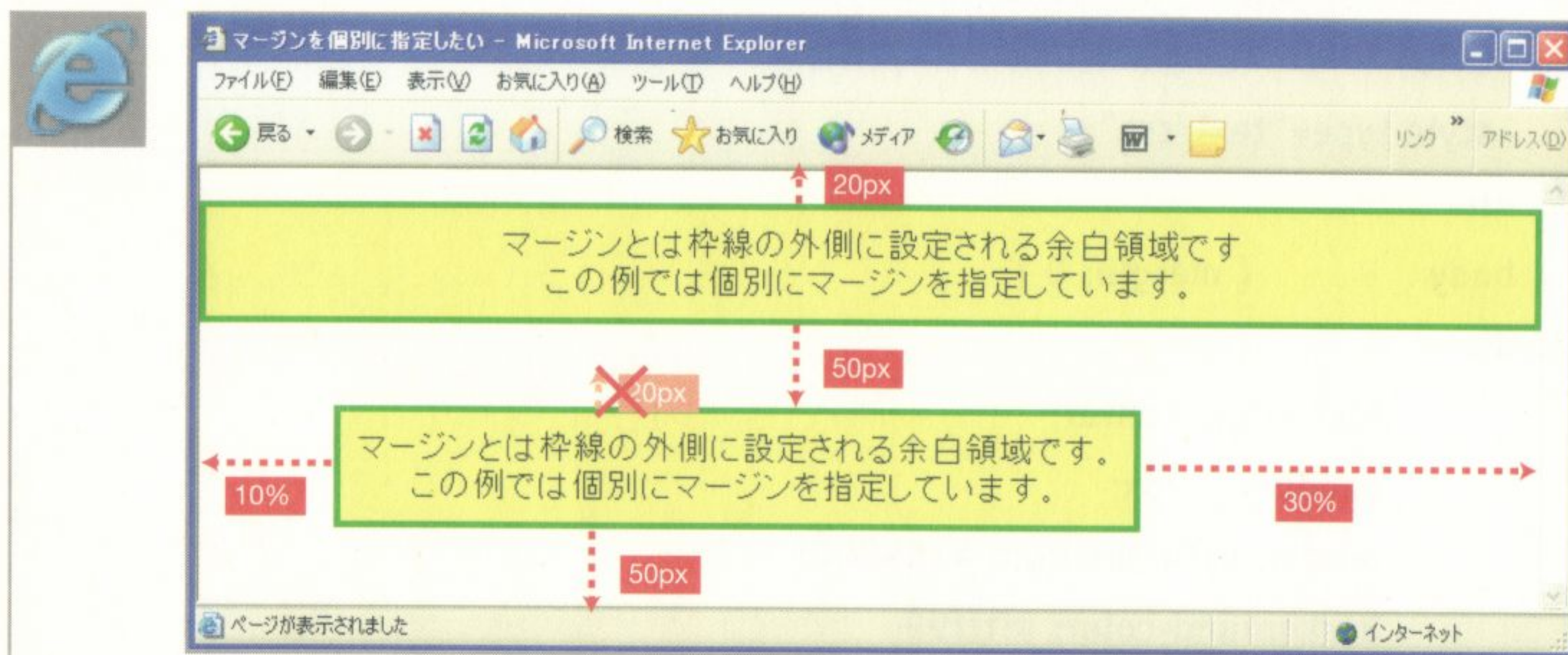
auto　　ブラウザが自動的にマージンを設定

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>マージンを個別に指定したい</title>
<style type="text/css">
<!--
body		{ margin: 0 }
div		{
	text-align: center;
	padding: 10px;
	border: solid medium #339900;
	background-color: #ffff99
}
#sample1	{
	margin-top: 20px;
	margin-bottom: 50px
}
#sample2	{
	margin-top: 20px;
	margin-right: 30%;
	margin-bottom: 50px;
	margin-left: 10%
}
-->
</style>
</head>
```

ボックス

```
<body>
<div id="sample1">
マージンとは枠線の外側に設定される余白領域です<br>
この例では個別にマージンを指定しています。
</div>
<div id="sample2">
マージンとは枠線の外側に設定される余白領域です。<br>
この例では個別にマージンを指定しています。
</div>
</body>
</html>
```

マージンを指定するHTMLタグをCSSに改める

HTMLタグで実現できるマージンは、ページ全体に対するマージンのみとなります。次のように<body>タグに属性を指定します。

```
<body topmargin="★"> ～ </body>
<body rightmargin="★"> ～ </body>
<body bottommargin="★"> ～ </body>
<body leftmargin="★"> ～ </body>
```

★――マージンの値

HTMLタグのマージン指定はInternet Explorerの独自拡張で、Netscape（Navigator）は対応していませんが、スタイルシートならばNetscape（Navigator）も対応しています。

なお、スタイルシートを使えばページだけでなく、すべての要素にマージンを指定することができます。同様の効果をスタイルシートで表現すると一例として以下のようになります。

```
body { margin-top: ★ px }
body { margin-right: ★ px }
body { margin-bottom: ★ px }
body { margin-left: ★ px }
```

★――マージンの値

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○
auto	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

マージンを一括して指定したい……………………p.164

2 BOX

マージンを一括して指定したい

margin: ★	上下左右同じ
margin: ★ ★	上下、左右
margin: ★ ★ ★	上、左右、下
margin: ★ ★ ★ ★	上、右、下、左

★……サイズを表す数値＋単位
　　パーセントを表す数値＋%
　　キーワード

ボックス

ボックスの枠と隣接するほかのボックス領域との間隔（マージン）を一括して指定します。隣接するボックスがない場合は親要素との間隔が設定されます。

値が1つだけのときは上下左右に同じマージン幅が適用されますが、2～4個の値を半角スペースで区切って並べると、値の数によって適用されるマージン幅と場所の組み合わせが変わります。

なお、上下に隣接するブロックレベル要素同士のマージンは、相殺されて大きいほうのマージンが設定されます。

値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけてマージンを指定します。マイナスの数値を指定することもできます。単位についてはp.46を参照してください。

パーセントを表す数値＋%

親要素のボックス領域の幅に対する割合でマージンを指定します。左右だけでなく、上下のマージンについても幅を基準にします。

キーワード

auto	ブラウザが自動的にマージンを設定

SOURCE

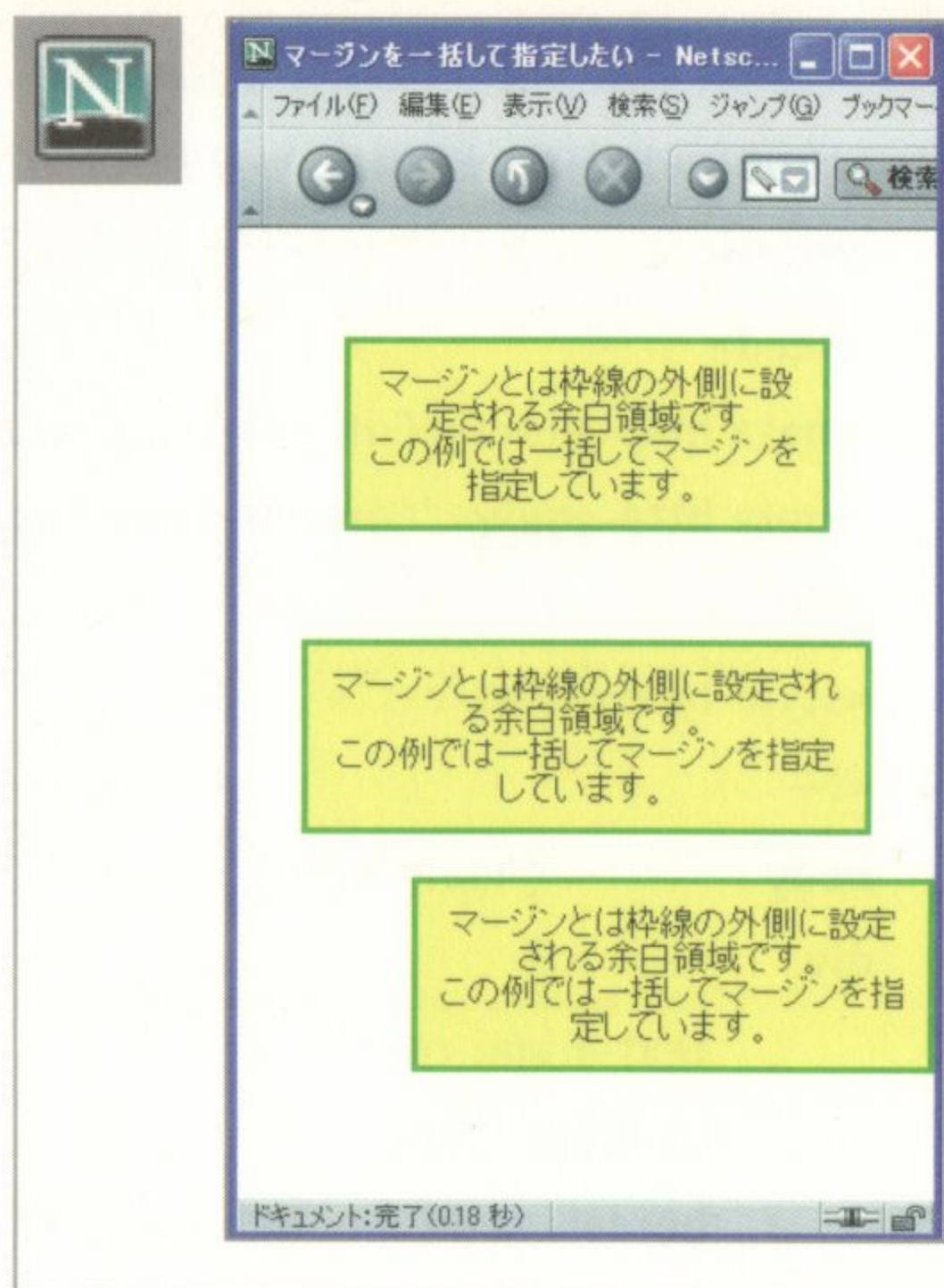
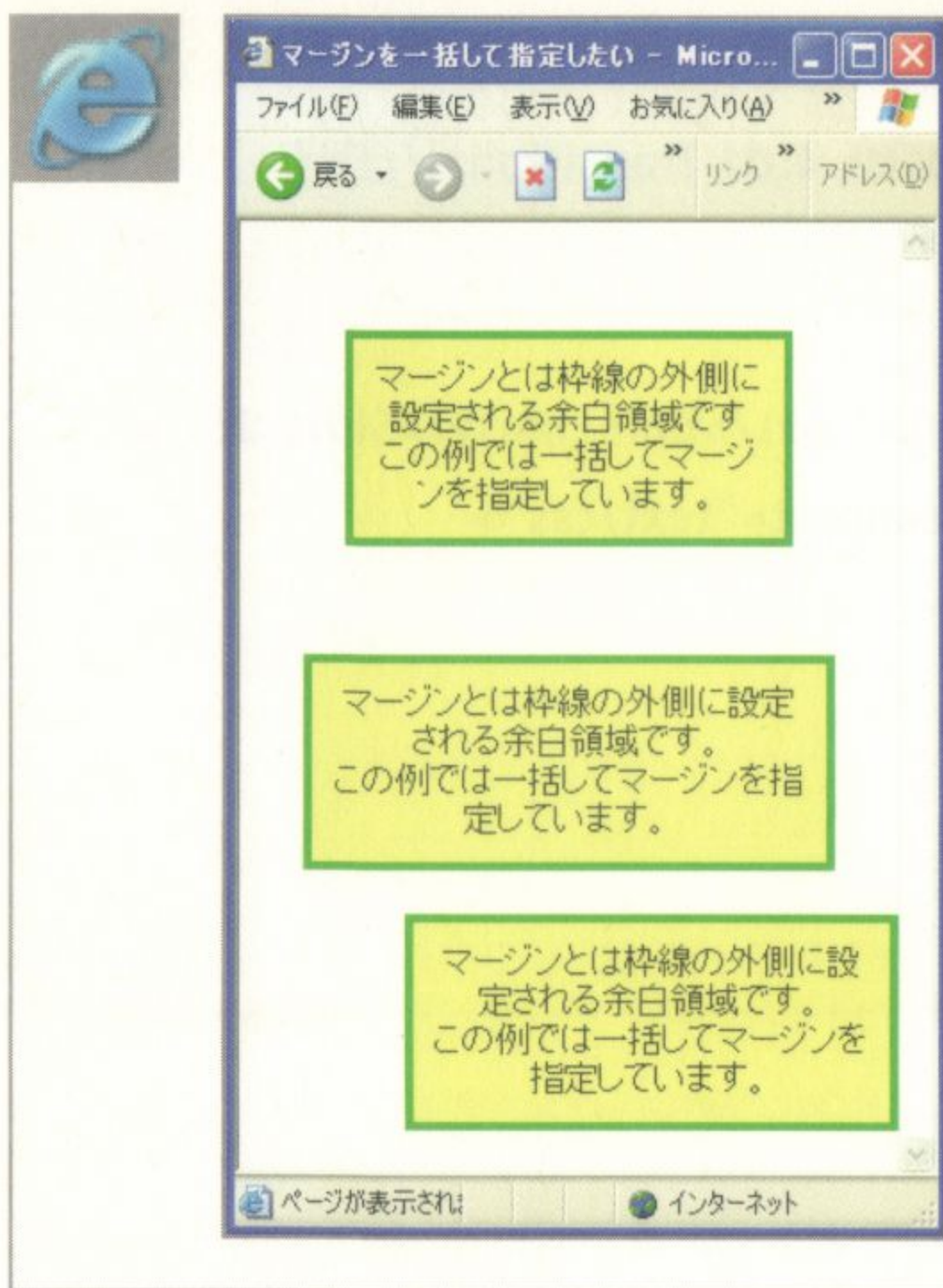
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>マージンを一括して指定したい</title>
<style type="text/css">
<!--
body		{ margin: 0 }
div		{
	text-align: center;
	padding: 10px;
	border: solid medium #339900;
	background-color: #ffff99
}
#sample1	{ margin: 50px }
#sample2	{ margin: 20px 30px }
#sample3	{ margin: 20px 0px 10px 25% }
-->
</style>
</head>
<body>
<div id="sample1">
マージンとは枠線の外側に設定される余白領域です<br>
この例では一括してマージンを指定しています。
</div>
<div id="sample2">
……（中略）……
</div>
<div id="sample3">
……（中略）……
</div>
</body>
</html>
```

ボックス

ボックス

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○
auto	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照 マージンを個別に指定したい……………………p.160

パディングを個別に指定したい

padding-top: ★ 上側
padding-right: ★ 右側
padding-bottom: ★ 下側
padding-left: ★ 左側

★……サイズを表す数値＋単位
パーセントを表す数値＋%
キーワード

ボックスの内容領域と枠との間の間隔（パディング）を個別に指定します。

padding-topは上側、padding-rightは右側、padding-bottomは下側、padding-leftは左側のパディングにそれぞれ適用されます。

値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけてパディングを指定します。単位についてはp.46を参照してください。マージンと違い、パディングにはマイナスを指定することはできません。

パーセントを表す数値＋%

親要素のボックス領域の幅に対する割合でパディングを指定します。左右だけでなく、上下のパディングについても幅を基準にします。

キーワード

auto	ブラウザが自動的にパディングを設定

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>パディングを個別に指定したい</title>
<style type="text/css">
<!--
body            { margin: 0 }
div             {
        margin: 20px;
        border: solid medium #339900;
        background-color: #ffff99
}
#sample1        {
        padding-top: 20px;
        padding-bottom: 50px
}
#sample2        {
        padding-top: 20px;
        padding-right: 30%;
        padding-bottom: 50px;
        padding-left: 10%
}
-->
</style>
</head>
<body>
<div id="sample1">パディングとは……（中略）……います。</div>
<div id="sample2">パディングとは……（中略）……います。</div>
</body>
</html>
```

ボックス

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○
auto	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照
パディングを一括して指定したい……………p.170

4 BOX

パディングを一括して指定したい

padding: ★	上下左右同じ
padding: ★ ★	上下、左右
padding: ★ ★ ★	上、左右、下
padding: ★ ★ ★ ★	上、右、下、左

★……サイズを表す数値＋単位
　　パーセントを表す数値＋%
　　キーワード

ボックス

ボックスの内容領域と枠との間の間隔（パディング）を一括して指定します。

値が1つだけのときは上下左右に同じパディング幅が適用されますが、2～4個の値を半角スペースで区切って並べると、値の数によって適用されるパディング幅と場所の組み合わせが変わります。

値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけてパディングを指定します。単位についてはp.46を参照してください。マージンと違い、パディングにはマイナスを指定することはできません。

パーセントを表す数値＋%

親要素のボックス領域の幅に対する割合でパディングを指定します。左右だけでなく、上下のパディングについても幅を基準にします。

キーワード

auto	ブラウザが自動的にパディングを設定

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>パディングを一括して指定したい</title>
<style type="text/css">
<!--
body          { margin: 0 }
div           {
        margin: 20px;
        border: solid medium #339900;
        background-color: #ffff99
}
#sample1      { padding: 50px }
#sample2      { padding: 20px 30px }
#sample3      { padding: 20px 0px 10px 25% }
-->
</style>
</head>
<body>
<div id="sample1">
パディングとは要素の内容が表示される部分と枠線との間の余白領域です。
この例では上下左右に50ピクセルのパディングが設定されています。
</div>
<div id="sample2">
パディングとは……（中略）……います。
</div>
<div id="sample3">
パディングとは……（中略）……います。
</div>
</body>
</html>
```

ボックス

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○
auto	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

パディングを個別に指定したい……………………p.167

5 BOX

枠線の太さを個別に指定したい

border-top-width: ★ 上側

border-right-width: ★ 右側

border-bottom-width: ★ 下側

border-left-width: ★ 左側

★••••••サイズを表す数値＋単位
キーワード

ボックス

ボックス領域の上下左右の枠線（ボーダー）の太さを個別に指定します。

border-top-width は上の枠線に、border-right-width は右の枠線に、borer-bottom-width は下の枠線に、border-left-width は左の枠線にそれぞれ適用されます。

値には次のような指定方法があります。

なお、Netscape Navigator 4.7 以前では、この border-width プロパティを省略すると、border-style などを指定しても枠線は表示されません。

サイズを表す数値＋単位

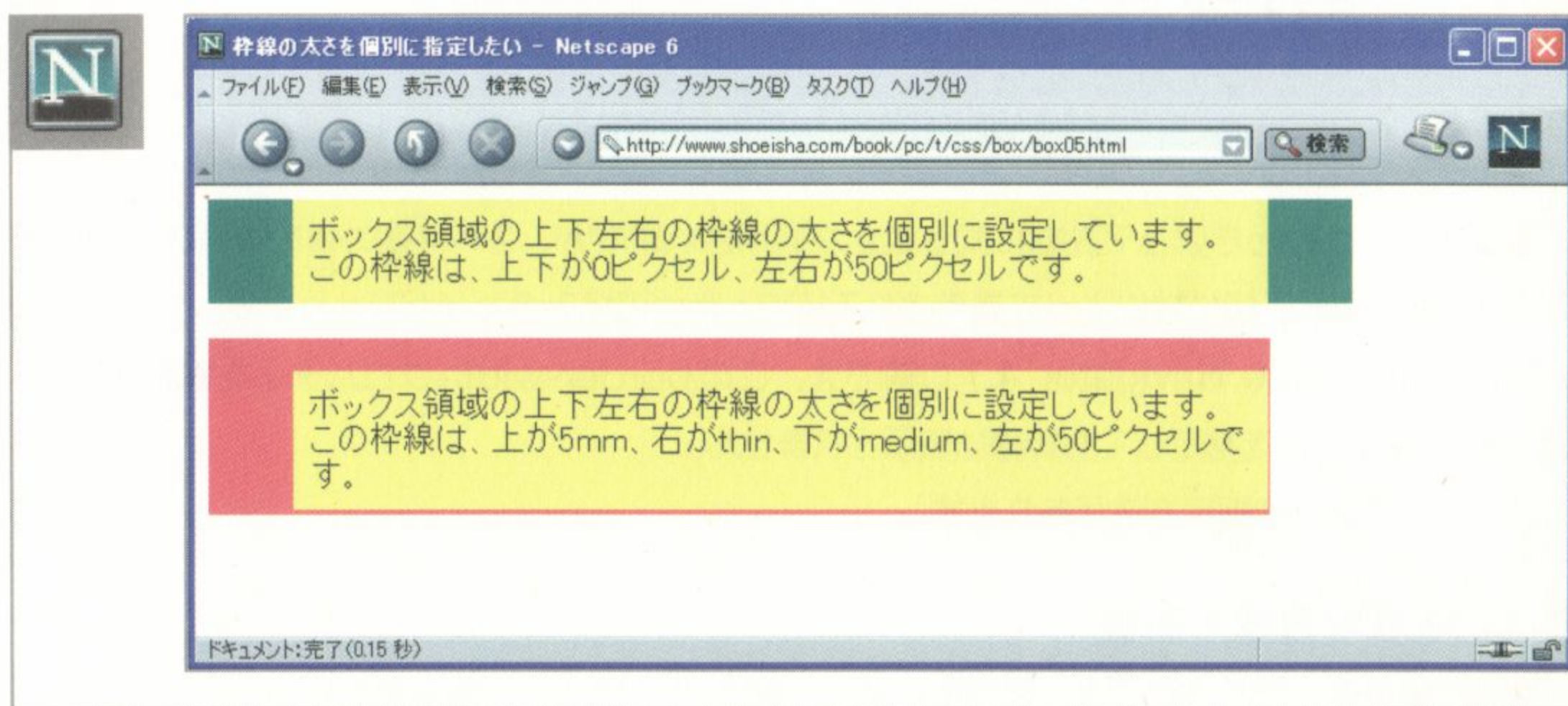
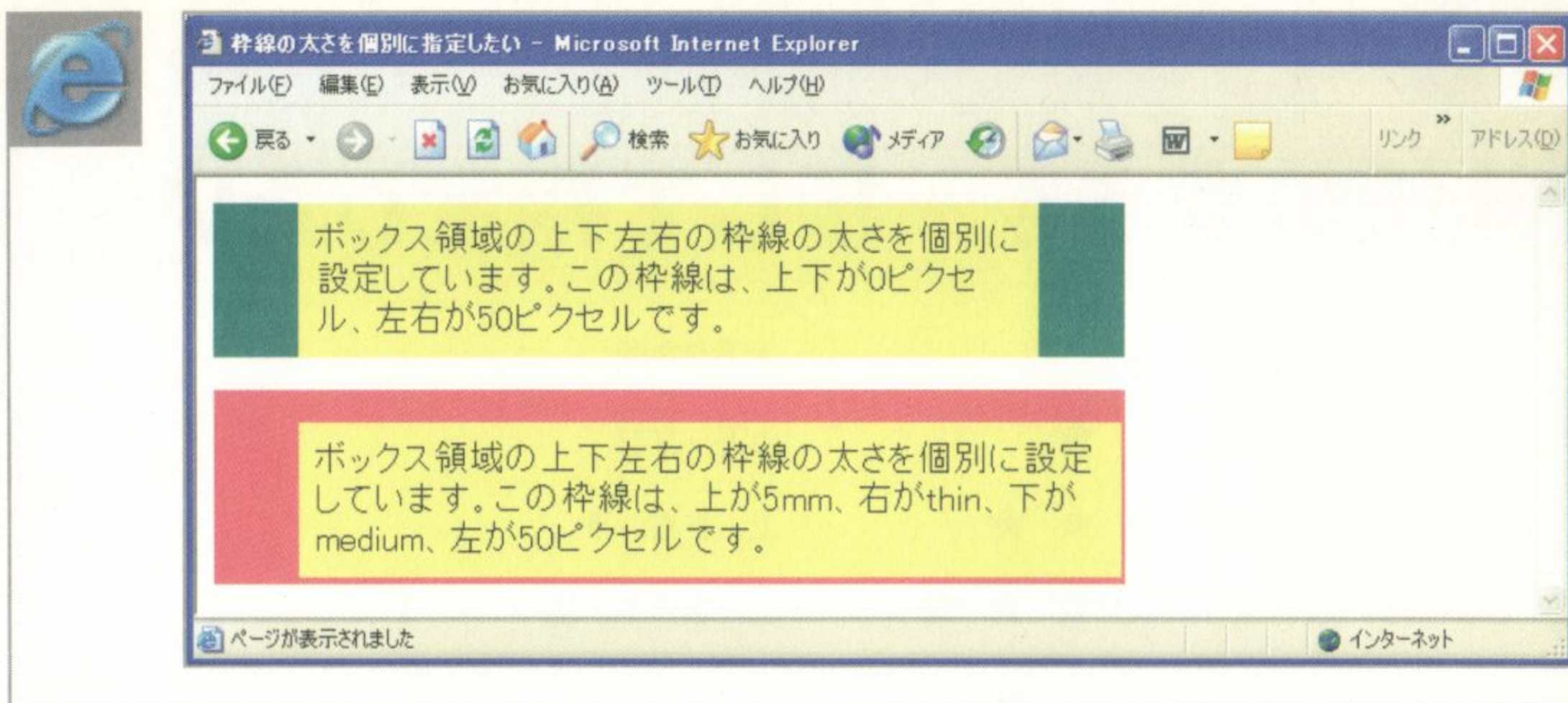
数値に単位をつけて枠線の太さを指定します。単位については p.46 を参照してください。

キーワード

thin	細い線
medium	中くらいの線（デフォルト）
thick	太い線

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title> 枠線の太さを個別に指定したい </title>
<style type="text/css">
```

```
<!--
p            {
        width: 70%;
        padding: 10px;
        background-color: #ffff99
}
.sample1      {
        border-top-width: 0px;
        border-right-width: 50px;
        border-bottom-width: 0px;
        border-left-width: 50px;
        border-style: solid;
        border-color: teal
}
.sample2      {
        border-top-width: 5mm;
        border-right-width: thin;
        border-bottom-width: medium;
        border-left-width: 50px;
        border-style: solid;
        border-color: #ff6666
}
-->
</style>
</head>
<body>
<p class="sample1">
ボックス領域の上下左右の枠線の太さを個別に指定しています。この枠線は、上下が0ピクセル、左右が50ピクセルです。
</p>
<p class="sample2">
ボックス領域の上下左右の枠線の太さを個別に指定しています。この枠線は、上が5mm、右がthin、下がmedium、左が50ピクセルです。
</p>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
thin	○	○	○	○	○	○	○
medium	○	○	○	○	○	○	○
thick	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照
枠線の太さを一括して指定したい……………p.176
個別の枠線ごとに一括して指定したい…………p.192
枠線を一括して指定したい……………………p.195

6 BOX

枠線の太さを一括して指定したい

border-width: ★	上下左右同じ
border-width: ★ ★	上下、左右
border-width: ★ ★ ★	上、左右、下
border-width: ★ ★ ★ ★	上、右、下、左

★……サイズを表す数値＋単位
　　　キーワード

ボックス

ボックス領域の上下左右の枠線（ボーダー）の太さを一括して指定します。

値が1つだけのときは上下左右に同じ太さが適用されますが、2～4個の値を半角スペースで区切って並べると、値の数によって適用される枠線と太さの組み合わせが変わります。

なお、Netscape Navigator 4.7以前では、このborder-widthプロパティを省略すると、border-styleなどを指定しても枠線は表示されません。

値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけて枠線の太さを指定します。単位についてはp.46を参照してください。

キーワード

thin	細い線
medium	中くらいの線（デフォルト）
thick	太い線

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>枠線の太さを一括して指定したい</title>
<style type="text/css">
<!--
p		{
	width: 70%;
	padding: 10px;
	background-color: #ffff99
}
.sample1	{
	border-width: medium;
	border-style: solid;
	border-color: teal
}
.sample2	{
	border-width: 30px 3px;
	border-style: solid;
	border-color: #ff6666
}
-->
</style>
</head>
<body>
<p class="sample1">ボックス領域の……（中略）……です。</p>
<p class="sample2">ボックス領域の……（中略）……です。</p>
</body>
</html>
```

ボックス

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
thin	○	○	○	○	○	○	○
medium	○	○	○	○	○	○	○
thick	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

枠線の太さを個別に指定したい……………………p.173
個別の枠線ごとに一括して指定したい…………p.192
枠線を一括して指定したい…………………………p.195

7 BOX

枠線の色を個別に指定したい

border-top-color: ★ 上側
border-right-color: ★ 右側
border-bottom-color: ★ 下側
border-left-color: ★ 左側

★••••••RGB値
キーワード
transparent（透明）

ボックス

ボックス領域の上下左右の枠線（ボーダー）の色を個別に指定します。

border-top-colorは上の枠線に、border-right-colorは右の枠線に、borer-bottom-colorは下の枠線に、border-left-colorは左の枠線にそれぞれ適用されます。

色はRGB値、キーワード、transparent（透明）のいずれかで指定します。transparentを指定すると、枠線の幅は確保されたまま透明で表示され、背景の色が透過されるようになります。色の詳しい指定方法についてはp.47を参照してください。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>枠線の色を個別に指定したい</title>
<style type="text/css">
<!--
p            {
        width: 70%;
        padding: 15px;
        border: solid 10px;
        text-align: center
}
```

```
.sample1        {
        border-top-color: red;
        border-right-color: yellow;
        border-bottom-color: green;
        border-left-color: blue
}
.sample2        {
        border-top-color: teal;
        border-bottom-color: navy
}
-->
</style>
</head>
<body>
<p class="sample1">
ボックス領域の上下左右の枠線の色を個別に指定しています。
</p>
<p class="sample2">
ボックス領域の上下左右の枠線の色を個別に指定しています。
</p>
</body>
</html>
```

ボックス

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
色名	×	○	○	○	×	×	○
システムカラー	×	○	○	○	×	×	○
transparent	×	○	○	○	×	×	○
#rgb	×	○	○	○	×	×	○
#rrggbb	×	○	○	○	×	×	○
rgb(%,%,%)	×	○	○	○	×	×	○
rgb(,,)	×	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

参照
枠線の色を一括して指定したい…………………p.182
個別の枠線ごとに一括して指定したい…………p.192
枠線を一括して指定したい……………………p.195

8 BOX

枠線の色を一括して指定したい

border-color: ★	上下左右同じ
border-color: ★ ★	上下、左右
border-color: ★ ★ ★	上、左右、下
border-color: ★ ★ ★ ★	上、右、下、左

★……RGB値
キーワード
transparent（透明）

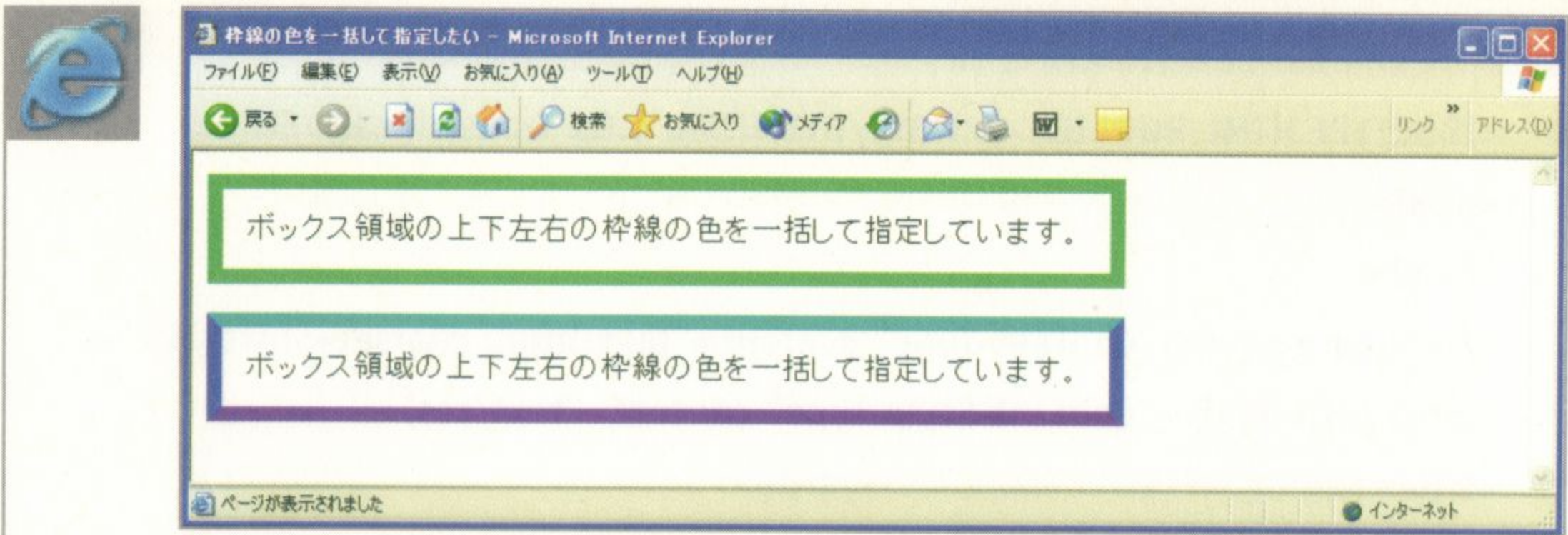
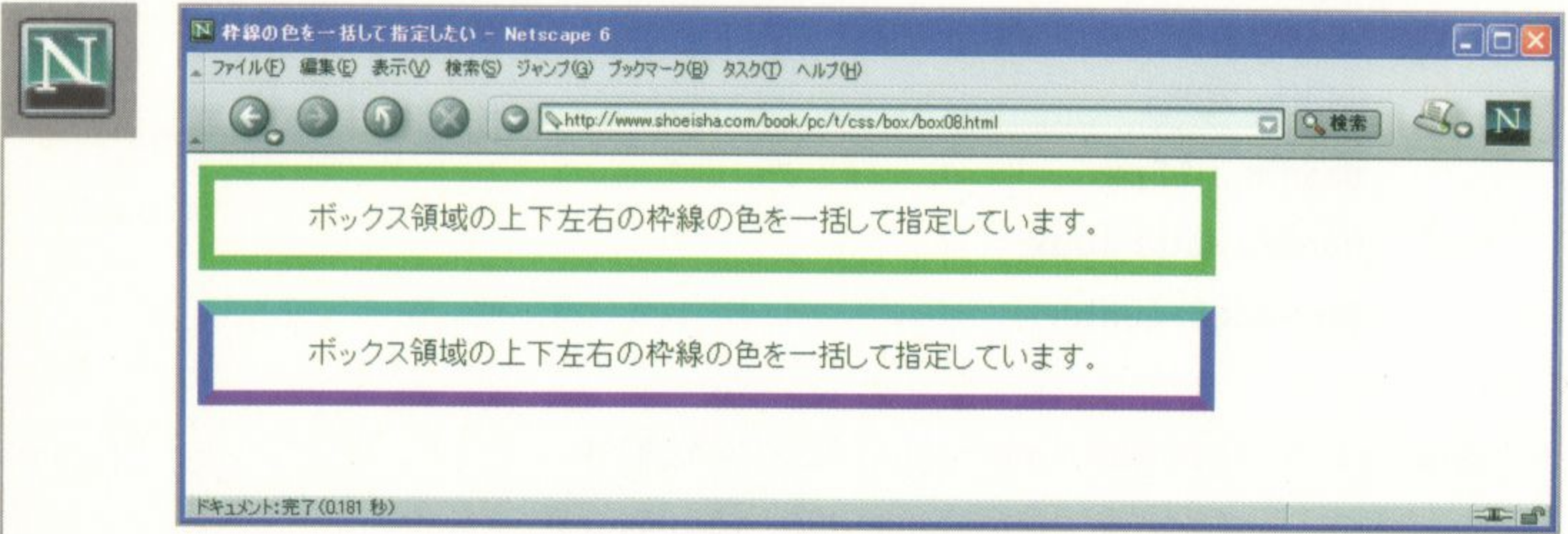
ボックス領域の上下左右の枠線（ボーダー）の色を一括して指定します。

値が1つだけのときは上下左右に同じ色が適用されますが、2～4個の値を半角スペースで区切って並べると、値の数によって適用される枠線と色の組み合わせが変わります。

色はRGB値、キーワード、transparent（透明）のいずれかで指定します。transparentを指定すると、枠線の幅は確保されたまま透明で表示され、背景の色が透過されるようになります。色の詳しい指定方法についてはp.47を参照してください。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>枠線の色を一括して指定したい</title>
<style type="text/css">
<!--
p		{
	width: 70%;
	padding: 15px;
	border: solid 10px;
	text-align: center
}
.sample1	{ border-color: rgb(102,204,153) }
.sample2	{ border-color: aqua #00f #800080 }
-->
</style>
</head>
<body>
<p class="sample1">
ボックス領域の上下左右の枠線の色を一括して指定しています。
</p>
<p class="sample2">
ボックス領域の上下左右の枠線の色を一括して指定しています。
</p>
</body>
</html>
```

ボックス

▲値が１つのときは上下左右に同じ色が、値が３つのときは左右が同じ色となります

▲値が１つのときは上下左右に同じ色が、値が３つのときは左右が同じ色となります

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
色名	○	○	○	○	○	○	○
システムカラー	○	○	○	○	×	×	○
transparent	×	×	×	×	×	×	○
#rgb	○	○	○	○	○	○	○
#rrggbb	○	○	○	○	○	○	○
rgb(%,%,%)	○	○	○	○	○	○	○
rgb(,,)	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照
枠線の色を一括して指定したい…………………p.182
個別の枠線ごとに一括して指定したい…………p.192
枠線を一括して指定したい……………………p.195

枠線の種類を個別に指定したい

border-top-style: ★	上側
border-right-style: ★	右側
border-bottom-style: ★	下側
border-left-style: ★	左側

★……キーワード

ボックス領域の上下左右の枠線（ボーダー）の種類を個別に指定します。

border-top-styleは上の枠線に、border-right-styleは右の枠線に、border-bottom-styleは下の枠線に、border-left-styleは左の枠線にそれぞれ適用されます。

値には以下のキーワードがあります（それぞれの実例はp.190参照）。

none	枠線を表示しない（デフォルト）
hidden	枠線を表示しない
dotted	点線
dashed	破線
solid	実線
double	二重線
groove	線がへこんだように見える枠線
ridge	線が浮き上がったように見える枠線
inset	線より内側がへこんだように見える枠線
outset	線より内側が浮き上がったように見える枠線

border-styleプロパティを省略するとデフォルトのnoneが適用され、枠線は表示されません。

また、noneとhiddenはどちらも枠線を表示せず、太さも0に指定される点では共通していますが、テーブルのセルなどの枠線として重なりあった場合にはnoneは他の値を優先し、hiddenは自分の値を優先します（p.206参照）。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>枠線の種類を個別に指定したい</title>
<style type="text/css">
<!--
p       {
        text-align: center;
        font-weight: bold;
        width: 70%;
        padding: 15px;
        border: 7px rgb(102,204,153);
        border-top-style: dotted;
        border-right-style: solid;
        border-bottom-style: dashed;
        border-left-style: double
}
-->
</style>
</head>
<body>
<p>
border-top-style: dotted<br>
border-right-style: solid<br>
border-bottom-style: dashed<br>
border-left-style: double
</p>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
none	○	○	○	○	×	×	○
dotted	×	×	○	○	×	×	○
dashed	×	×	○	○	×	×	○
solid	○	○	○	○	×	×	○
double	○	○	○	○	×	×	○
groove	○	○	○	○	×	×	○
ridge	○	○	○	○	×	×	○
inset	○	○	○	○	×	×	○
outset	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります
※Macintosh版IE5はdotted、dashedにも対応しています

枠線の種類を一括して指定したい……………p.188
個別の枠線ごとに一括して指定したい…………p.192
枠線を一括して指定したい……………………p.195

10 BOX

枠線の種類を一括して指定したい

border-style: ★ 上下左右同じ

border-style: ★ ★ 上下、左右

border-style: ★ ★ ★ 上、左右、下

border-style: ★ ★ ★ ★ 上、右、下、左

★……キーワード

ボックス領域の上下左右の枠線（ボーダー）の種類を一括して指定します。

値が1つだけのときは上下左右に同じ種類が適用されますが、2～4個の値を半角スペースで区切って並べると、値の数によって適用される枠線と種類の組み合わせが変わります。値には以下のキーワードがあります。

none	枠線を表示しない（デフォルト）
hidden	枠線を表示しない
dotted	点線
dashed	破線
solid	実線
double	二重線
groove	線がへこんだように見える枠線
ridge	線が浮き上がったように見える枠線
inset	線より内側がへこんだように見える枠線
outset	線より内側が浮き上がったように見える枠線

border-styleプロパティを省略するとデフォルトのnoneが適用され、枠線は表示されません。

また、noneとhiddenはどちらも枠線を表示せず、太さも0に指定される点では共通していますが、テーブルのセルなどの枠線として重なりあった場合にはnoneは他の値を優先し、hiddenは自分の値を優先します（p.206参照）。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>枠線の種類を一括して指定したい</title>
<style type="text/css">
<!--
p            {
        width: 70%;
        padding: 10px;
        border: 7px rgb(102,204,153);
        text-align: center;
        font-weight: bold
}
p.sample1    { border-style: none }
p.sample2    { border-style: dotted }
p.sample3    { border-style: dashed }
p.sample4    { border-style: solid }
p.sample5    { border-style: double }
p.sample6    { border-style: groove }
p.sample7    { border-style: ridge }
p.sample8    { border-style: inset }
p.sample9    { border-style: outset }
p.sample10   { border-style: dotted double }
-->
</style>
</head>
<body>
<p class="sample1">border-style: none</p>
<p class="sample2">border-style: dotted</p>
<p class="sample3">border-style: dashed</p>
<p class="sample4">border-style: solid</p>
<p class="sample5">border-style: double</p>
```

ボックス

```
<p class="sample6">border-style: groove</p>
<p class="sample7">border-style: ridge</p>
<p class="sample8">border-style: inset</p>
<p class="sample9">border-style: outset</p>
<p class="sample10">border-style: dotted double</p>
</body>
</html>
```

ボックス

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
none	○	○	○	○	○	○	○
dotted	×	×	○	○	×	×	○
dashed	×	×	○	○	×	×	○
solid	○	○	○	○	○	○	○
double	○	○	○	○	○	○	○
groove	○	○	○	○	○	○	○
ridge	○	○	○	○	○	○	○
inset	○	○	○	○	○	○	○
outset	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

※ Macintosh版IE5はdotted、dashedにも対応しています

枠線の種類を個別に指定したい・・・・・・・・・・・・・・・・・・p.185
個別の枠線ごとに一括して指定したい・・・・・・・・・・・・p.192
枠線を一括して指定したい・・・・・・・・・・・・・・・・・・・・・・p.195

11 BOX

個別の枠線ごとに一括して指定したい

border-top: ★ ☆ ▲ 上側

border-right: ★ ☆ ▲ 右側

border-bottom: ★ ☆ ▲ 下側

border-left: ★ ☆ ▲ 左側

★••••••border-width の値（枠線の太さ）
☆••••••border-style の値（枠線の種類）
▲••••••border-color の値（枠線の色）

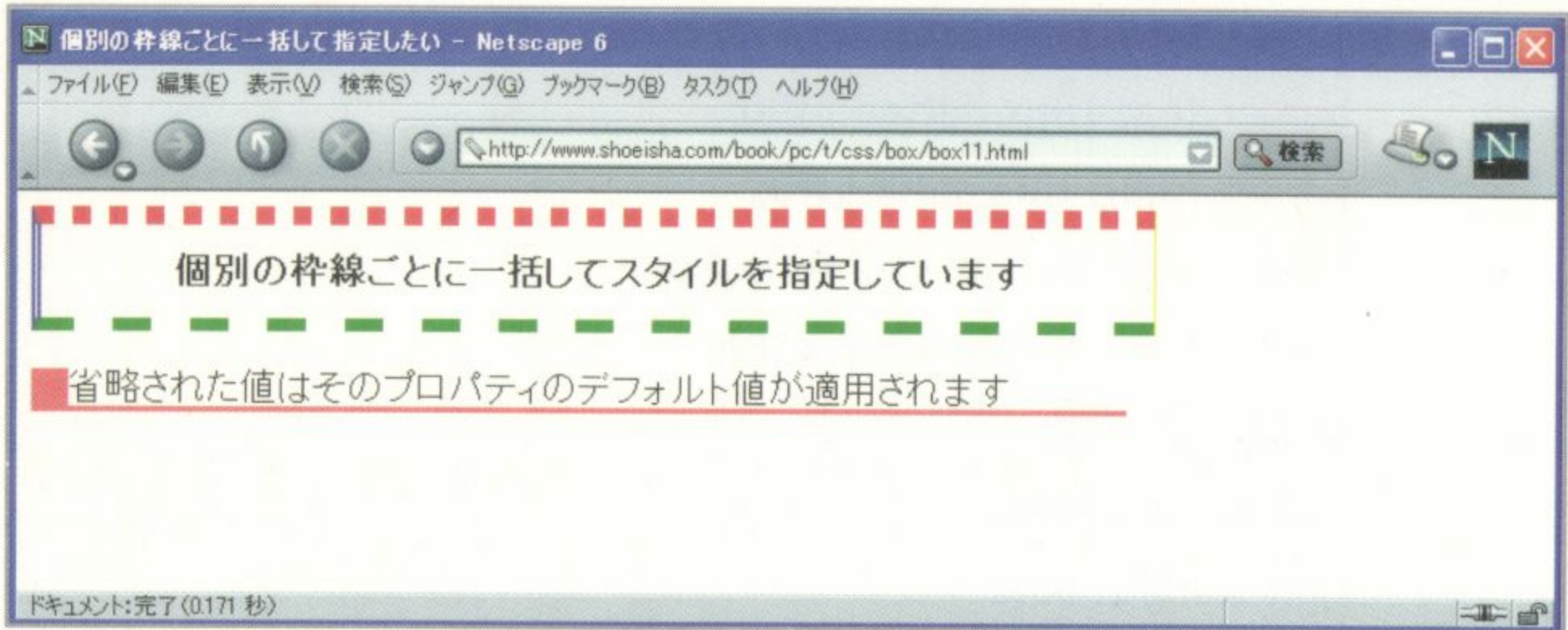
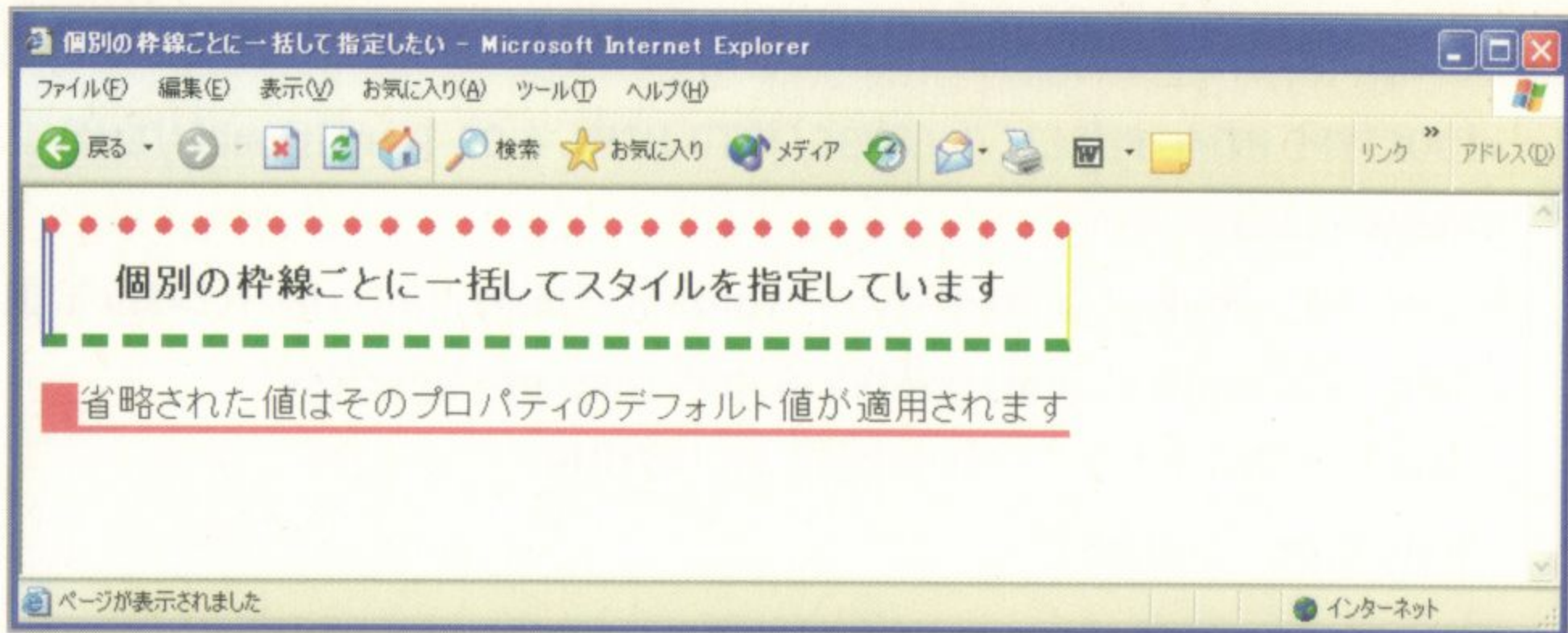
ボックス

ボックス領域の上下左右の枠線（ボーダー）の各種指定を、枠線ごとにまとめて指定します。

border-top は上の枠線に、border-right は右の枠線に、borer-bottom は下の枠線に、border-left は左の枠線に適用されます。

これらのプロパティに対して、太さ（border-width）、種類（border-style）、色（border-color）のそれぞれの値を、半角スペースで区切って任意の順番で指定します。値を省略すると各プロパティのデフォルトが適用されますので、border-style（デフォルトは none）の値は必ず指定してください。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>個別の枠線ごとに一括して指定したい</title>
<style type="text/css">
<!--
p               { width: 70% }
#sample1        {
        border-top: 10px dotted #f00;
        border-right: thin solid #ff0;
        border-bottom: 7px dashed #080;
        border-left: thick double #00f;
        padding: 15px;
        text-align: center;
        font-weight: bold
}
#sample2        {
        border-bottom: solid #f66;
        border-left: 20px solid #f66;
        text-align: left
}
-->
</style>
</head>
<body>
<p id="sample1">
個別の枠線ごとに一括してスタイルを指定しています
</p>
<p id="sample2">
省略された値はそのプロパティのデフォルト値が適用されます
</p>
</body>
</html>
```

ボックス

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
共通	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

参照
枠線の太さを個別に指定したい……………………p.173
枠線の色を個別に指定したい……………………p.179
枠線の種類を個別に指定したい…………………p.185
枠線を一括して指定したい………………………p.195

枠線を一括して指定したい

border: ★ ☆ ▲

★••••••border-width の値（枠線の太さ）
☆••••••border-style の値（枠線の種類）
▲••••••border-color の値（枠線の色）

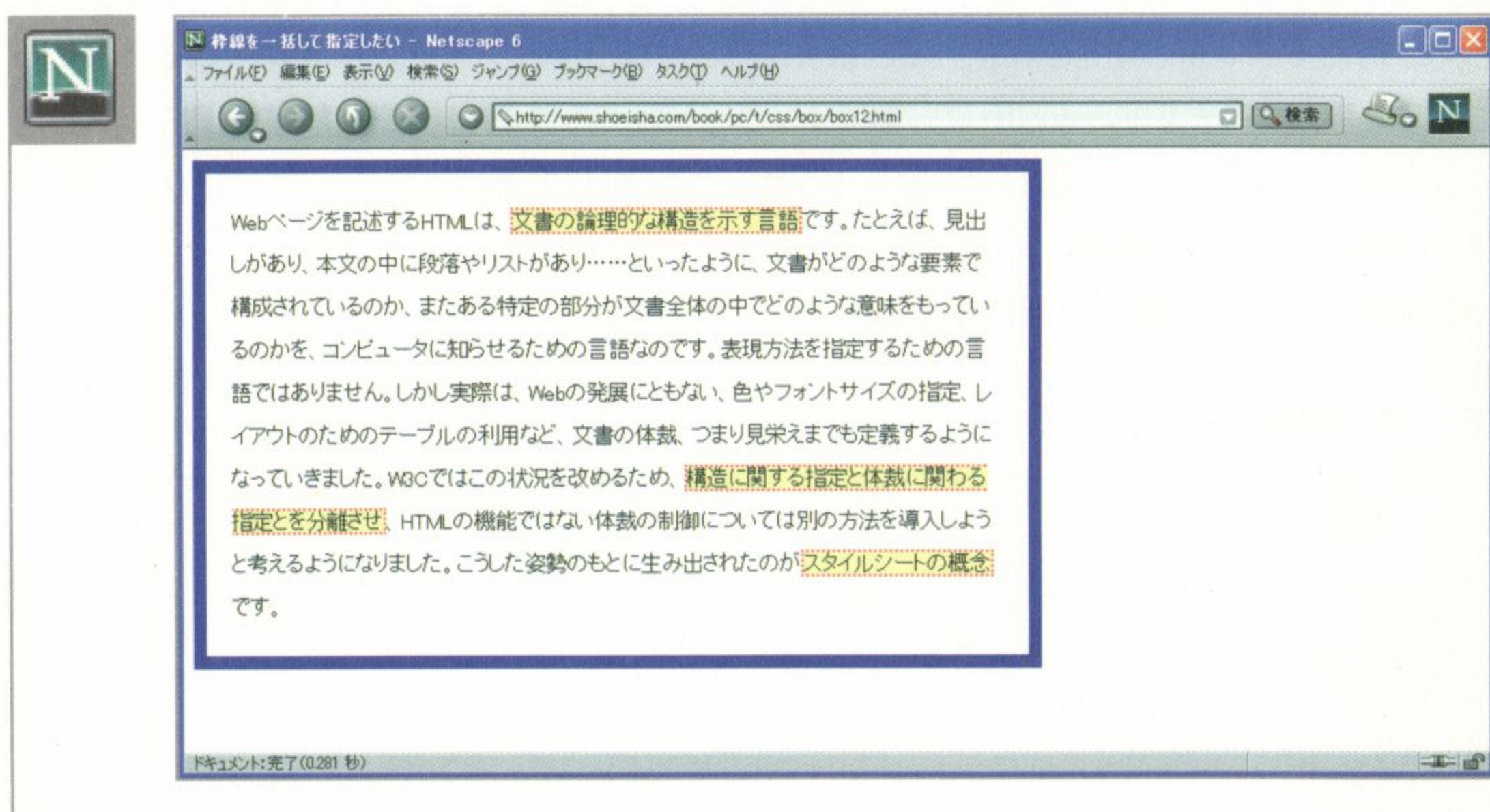
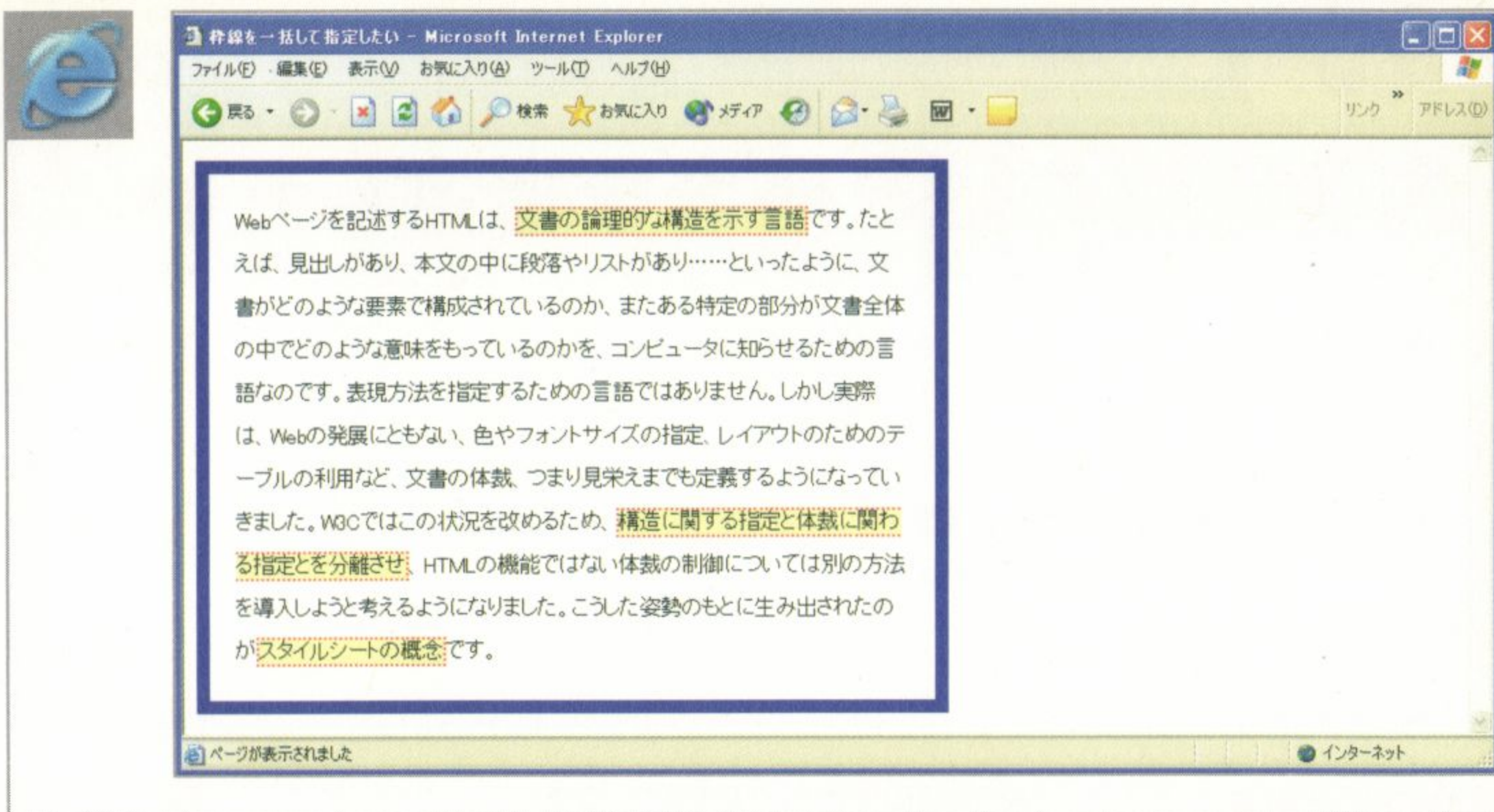
ボックス領域の上下左右の枠線（ボーダー）の各種指定をまとめて指定します。

太さ（border-width）、種類（border-style）、色（border-color）のそれぞれの値を、半角スペースで区切って任意の順番で指定します。値を省略すると各プロパティのデフォルトが適用されますので、border-style（デフォルトは none）の値は必ず指定してください。

なお、border プロパティでは上下左右の枠線に対して個別の指定を行うことはできません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title> 枠線を一括して指定したい </title>
<style type="text/css">
<!--
div     {
        font-size: 12pt;
        width: 60%;
        padding: 20px;
        border: 10px solid #000080;
        line-height: 2em
}
span    {
        border: 2px dotted #f06;
```

```
        background-color: #ffc
}
-->
</style>
</head>
<body>
<div>
Webページを記述するHTMLは、<span>文書の論理的な構造を示す言語</span>です。
たとえば、見出しがあり、本文の中に段落やリストがあり……といったように、文書がどの
ような要素で構成されているのか、またある特定の部分が文書全体の中でどのような意味を
もっているのかを、コンピュータに知らせるための言語なのです。表現方法を指定するため
の言語ではありません。しかし実際は、Webの発展にともない、色やフォントサイズの指
定、レイアウトのためのテーブルの利用など、文書の体裁、つまり見栄えまでも定義するよ
うになっていきました。W3Cではこの状況を改めるため、<span>構造に関する指定と体
裁に関わる指定とを分離させ</span>、HTMLの機能ではない体裁の制御については別の
方法を導入しようと考えるようになりました。こうした姿勢のもとに生み出されたのが
<span>スタイルシートの概念</span>です。
</div>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
共通	○	○	○	○	○	×	○

※適用するセレクタによっても効果が変わります

枠線の太さを一括して指定したい………………p.176
枠線の色を一括して指定したい…………………p.182
枠線の種類を一括して指定したい………………p.188
個別の枠線ごとに一括して指定したい…………p.192

13 BOX

内容領域の幅と高さを指定したい

width: ★	幅
height: ★	高さ

★……サイズを表す数値＋単位
　　パーセントを表す数値＋%
　　キーワード

要素の内容が表示される、内容領域の幅と高さを指定します。

これらのプロパティはブロックレベル要素と置換要素（指定された内容で置換される要素。たとえばimg要素はsrc属性で指定された内容に置換される。他にinput、textarea、selectなど）に対して適用されますが、widthは非置換インラインレベル要素や、テーブルの横列と横列グループに関する要素（tr、thead、tbody、tfoot）、heightは非置換インラインレベル要素やテーブルの縦列に関するcol要素、colgroup要素には適用されません。

値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけて幅や高さを指定します。単位についてはp.46を参照してください。

パーセントを表す数値＋%

親要素の幅や高さに対する割合でサイズを指定します。

キーワード

auto	ブラウザが自動的にサイズを設定

ブラウザによる幅と高さの違い

Internet Explorerの標準準拠モードと互換モード（p.51参照）ではwidthプロパティとheightプロパティによって設定されるサイズが異なります。CSSの標準仕様にしたがって正しく表示をする標準準拠モードでは要素の内容領域のサイズとして解釈・表示されますが、従来のブラウザと同様の表示をする互換モードでは内容領域にその周りのパディングと枠線を加えたサイズとして解釈・表示されます。パディングや枠線のとりかたによっては表示に大きな違いが生じてしまうことがありますので、widthとheightでサイズを指定する場合には注意してください。

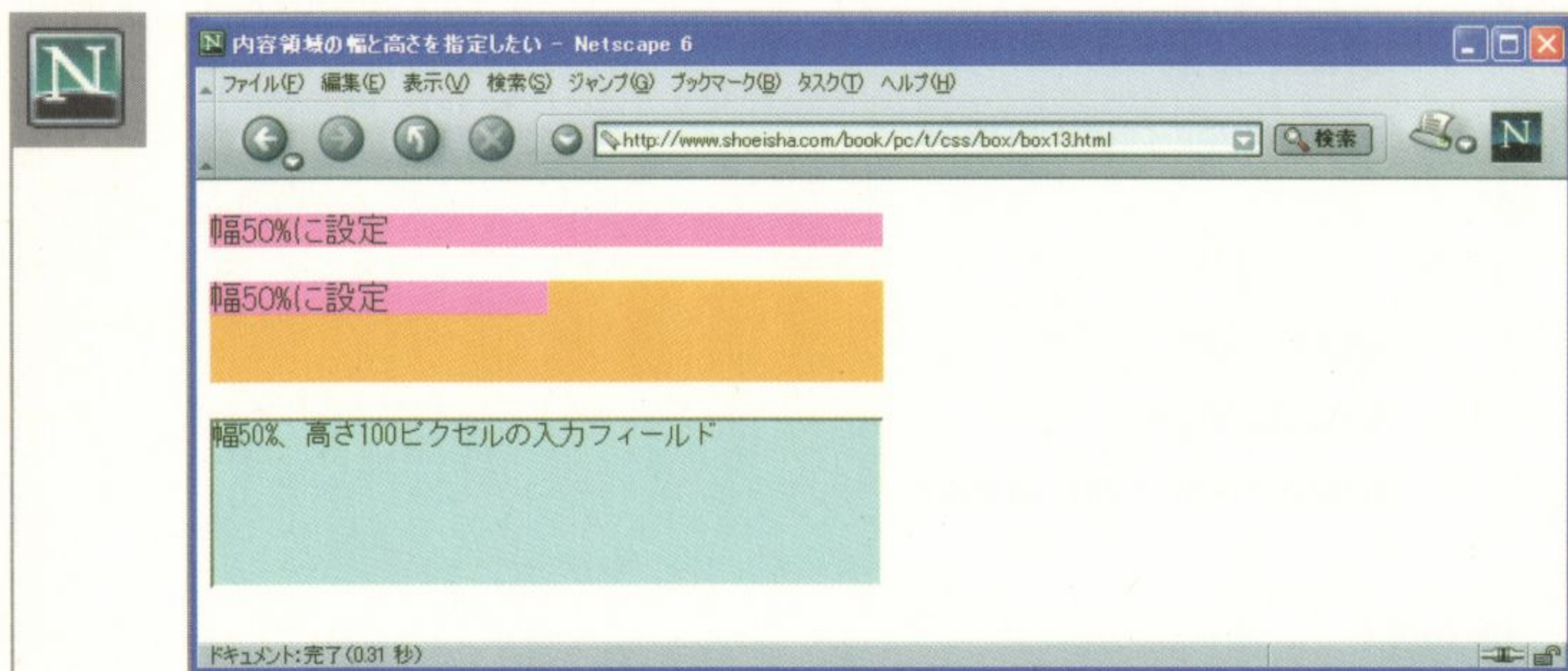
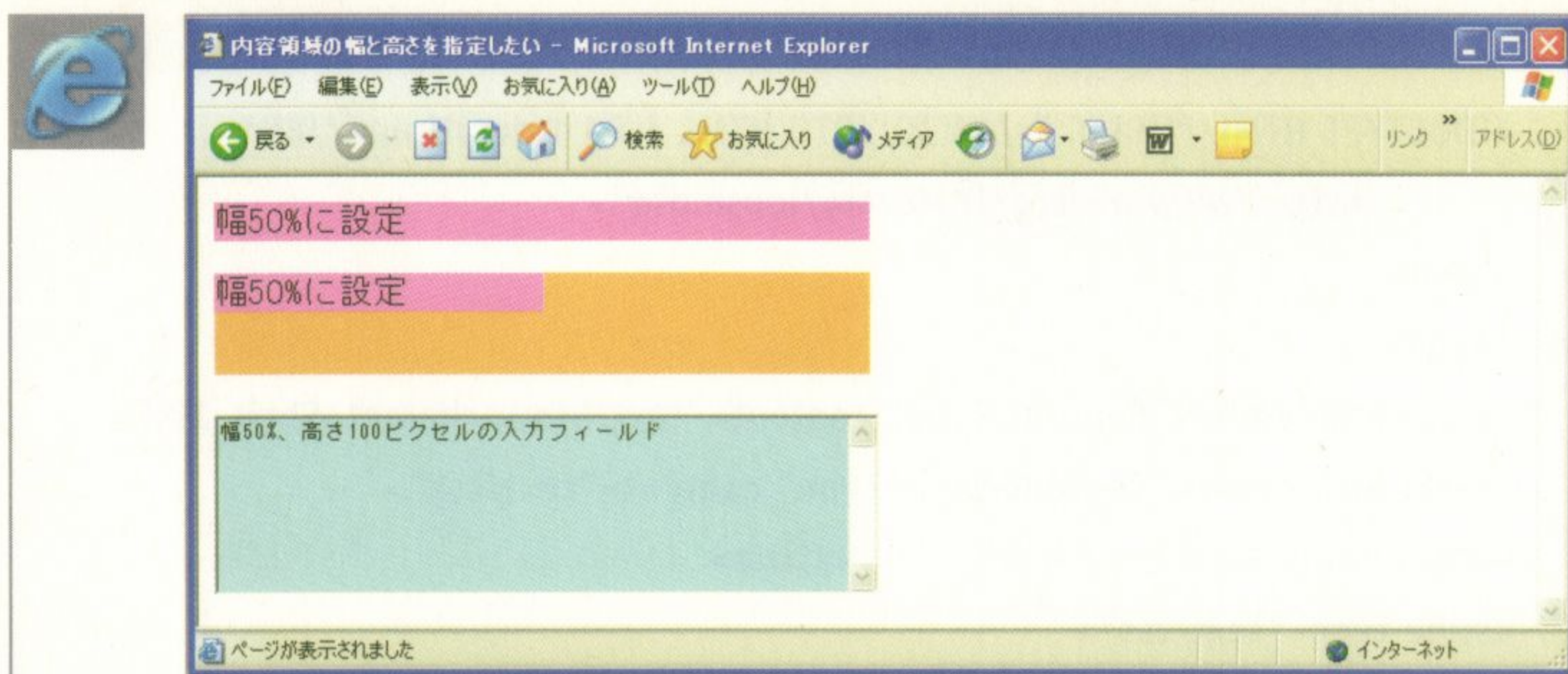
なお、Netscapeではどちらのモードであっても、内容領域のサイズとして解釈表示されます。p.175やp.197のサンプルで、両ブラウザの表示内容のサイズが大きく異なるのは、この違いによるものです。

本項のサンプルは標準準拠モードのため両ブラウザの表示の違いはありません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
        "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>内容領域の幅と高さを指定したい</title>
<style type="text/css">
<!--
p               {
        width: 50%;
        background-color: #ff99cc
}
div             {
        width: 50%;
        height: 60px;
        background-color: orange
}
textarea        {
        width: 50%;
        height: 100px;
        background-color: #ccffff
}

-->
</style>
</head>
<body>
<p>幅50%に指定</p>
<div><p>幅50%に指定</p></div>
<br>
<textarea>幅50%、高さ100ピクセルの入力フィールド</textarea>
</body>
</html>
```

ボックス

サイズを指定するHTMLタグをCSSに改める

HTMLでは画像やテーブルなどのサイズを次のようにwidth属性、height属性で指定します。

<img src="☆" width="★" height="▲">	画像のサイズ（☆――画像ファイルのURL）
<table width="★" height="▲">～</table>	テーブルのサイズ
<th width="★" height="▲">～</th>	セルのサイズ
<td width="★" height="▲">～</td>	セルのサイズ

★――幅の値
▲――高さの値

<th><td>タグへのサイズの指定はDeprecated（推奨しない）とされており、セルのサイズはスタイルシートで指定することが推奨されています。なお、スタイルシートを使うと、画像やテーブルだけでなく他の多くの要素に対してサイズを指定することができます。同様の効果をスタイルシートで表現すると以下のようになります。

```
img    { width: ★; height: ▲ }
table  { width: ★; height: ▲ }
th     { width: ★; height: ▲ }
td     { width: ★; height: ▲ }
```

★――幅の値
▲――高さの値

ボックス

width	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○
auto	○	○	○	○	○	○	○

height	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	×	×	○
パーセント	○	○	○	○	○	×	○
auto	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

内容があふれる場合の処理方法を指定したい……p.230

1 POSITIONING

表示形式を指定したい

display: ★

★……キーワード

要素をインラインレベル要素やブロックレベル要素に設定し、表示形式を指定します。
値には以下のキーワードがあります。

block	ブロックレベル要素として扱う
inline	インラインレベル要素として扱う（デフォルト）
list-item	リスト項目用のボックスを生成する
none	表示されない

blockを指定した要素はブロックレベル要素となり、要素の後に改行が入ります。プロパティの中にはブロックレベル要素にしか適用されないものもあるため、ブロックレベル以外の要素にそうしたプロパティを設定したい場合にはblockを指定します。

inlineを指定した要素はインラインレベル要素となり、要素の後に改行は入りません。list-itemでは指定した要素に対し、リスト内容を表示するボックスとリストマーク用のインラインボックスを生成させます。

noneを指定すると、要素が表示されなくなります。ボックスそのものが生成されないため、レイアウトにも影響を与えません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>表示形式を指定したい</title>
<style type="text/css">
<!--
span,div          { background-color: #66ff00 }
.sample1          { display: block }
.sample2          { display: none }
.sample3          { display: inline }
.sample4          { display: list-item }
div.sample5       { margin-left: 1em }
-->
</style>
</head>
<body>
<p>この下には表示されない画像があります。<br>
<img src="birds3.jpg" class="sample2" width="100" height="100" alt="">
</p>
<p>表示形式を指定します：<span class="sample1">blockを指定</span>するとこんな感じ。</p>
<p>表示形式を指定します：<span class="sample2">noneを指定</span>するとこんな感じ。</p>
<p>表示形式を指定します：<div class="sample3">inlineを指定</div>するとこんな感じ</p>
<p>表示形式を指定します：<div class="sample5"><span class="sample4">list-itemを指定</span><span class="sample4">list-itemを指定</span></div>するとこんな感じ。</p>
</body>
</html>
```

配置

表示形式を指定したい - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る 検索 お気に入り メディア リンク アドレス(D)
この下には表示されない画像があります。
表示形式を指定します:
blockを指定
するとこんな感じ。
表示形式を指定します: するとこんな感じ。
表示形式を指定します:
inlineを指定するとこんな感じ
表示形式を指定します:
list-itemを指定
list-itemを指定
するとこんな感じ。
ページが表示されました
インターネット

表示形式を指定したい - Netscape 6
ファイル(F) 編集(E) 表示(V) 検索(S) ジャンプ(G) ブックマーク(B) タスク(T) ヘルプ(H)
http://www.shoeisha.com/book/pc/t/css/positioning/positioning01.html
検索
この下には表示されない画像があります。
表示形式を指定します:
blockを指定
するとこんな感じ。
表示形式を指定します: するとこんな感じ。
表示形式を指定します:
inlineを指定するとこんな感じ
表示形式を指定します:
list-itemを指定
list-itemを指定
するとこんな感じ。
ドキュメント: 完了 (0.18 秒)

▲display プロパティを指定しない場合の表示

配置

display プロパティの用途

display プロパティでは、インラインレベル要素からブロックレベル要素へといった表示形式の変更が指定できます。ではこの属性はどのような場合に使用するのでしょうか。それは主にXML ということになります。

HTML における要素は、インラインレベル要素とブロックレベル要素に大別されますが（p.3 参照）、XML ではユーザーが定義しない限り要素はどちらにも属しません。この定義を行うのが display です。

同様に HTML での実装の仕方がわかりにくいプロパティとして、visible プロパティや z-index プロパティなどもあげることができます。しかしこれらのプロパティは Dynamic HTML などで Web ページを動的に動かす際にはなくてはならないものです。

スタイルシートは HTML だけではなく、Dynamic HTML や XML など Web テクノロジー全般にわたって使用できる技術なのです。

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
block	×	×	○	○	○	○	○
inline	×	○	○	○	○	○	○
list-item	×	×	○	○	○	○	○
none	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照 表示・非表示を指定したい……………………p.206

2 POSITIONING

表示・非表示を指定したい

visibility: ★

★……キーワード

要素の表示・非表示を指定します。
値には以下のキーワードがあります。

visible	表示（デフォルト）
hidden	表示しない

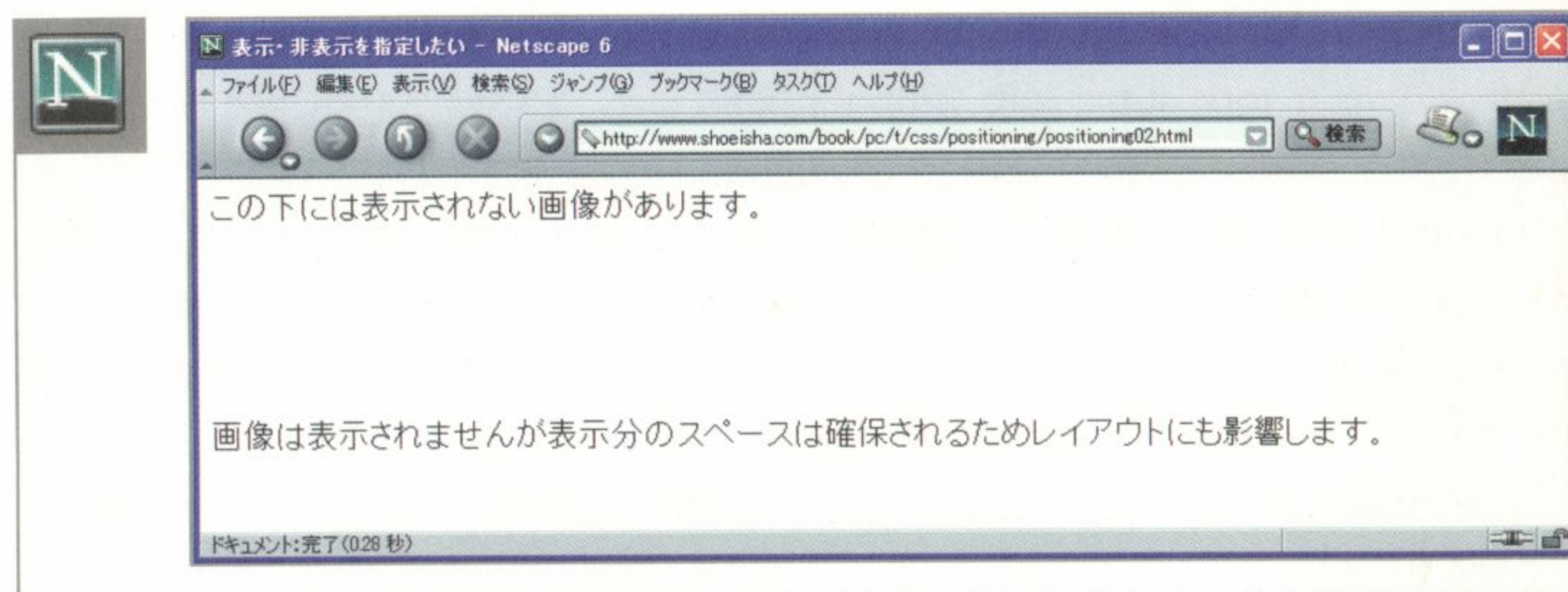
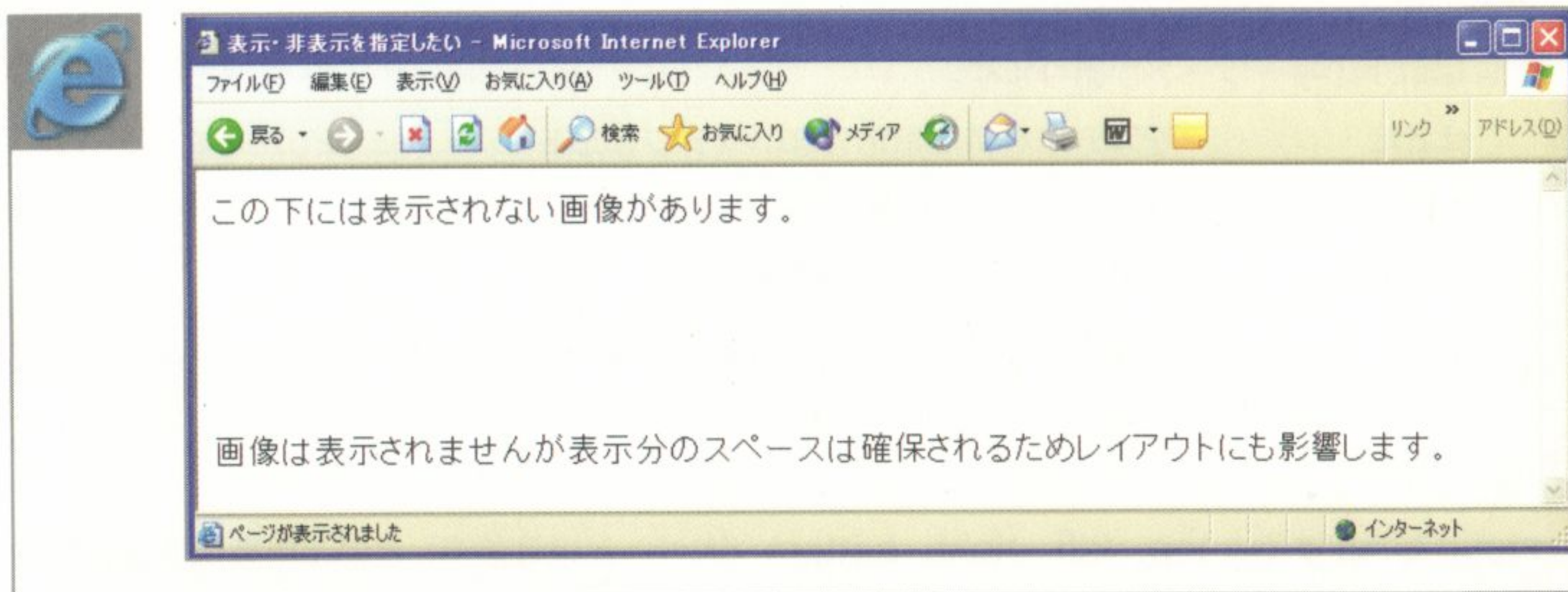
配置

display: none（p.202参照）はその要素そのものがないかのようにレイアウトにも影響を与えませんが、visibility: hiddenでは要素の内容は表示されませんが表示分のスペースは確保されるためレイアウトにも影響します。

visibilityプロパティでは、Dynamic HTMLやスクリプトと組み合わせてあるアクションによって画像が表示されるといった動的なページを、レイアウトを変更することなく作成することができます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>表示・非表示を指定したい</title>
<style type="text/css">
<!--
img    { visibility: hidden }
-->
</style>
</head>
```

```
<body>
<p> この下には表示されない画像があります。<br>
<img src="birds3.jpg" width="100" height="100" alt="">
</p>
<p>
画像は表示されませんが表示分のスペースは確保されるためレイアウトにも影響します。
</p>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
visible	○	○	○	○	×	×	○
hidden	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

表示形式を指定したい……………………………p.202
要素の一部を切り抜き表示したい………………p.225

3 POSITIONING

要素の配置方法を指定したい

position: ★

★……キーワード

要素の配置方法を指定します。
値には以下のキーワードがあります。

static	配置方法を特に指定しない（デフォルト）
relative	通常表示される位置からの相対的な配置
absolute	親要素に対して絶対的に配置
fixed	親要素に対して絶対的に配置（位置は固定され、スクロールしても移動しない）

staticを指定するとtop、left、right、bottom（次ページ以降参照）といった表示位置を調整するプロパティは無効になります。

relativeは、その要素が通常表示される位置からの相対的な配置を指定します。つまりrelativeの状態で"top: 40px; left: 120px"と指定すると、通常表示される位置を基準として、上端から40ピクセル、左端から120ピクセルの位置に表示されます。

absoluteは親要素からの絶対的な配置を指定します。たとえば、次ページのサンプルでは<div>タグの親要素である<body>タグのボックス領域上端から200ピクセル、左端から200ピクセルの位置に表示されます。この値を指定すると、他の要素に影響を与えない独立した配置になります。

fixedの配置位置はabsoluteと同じですが、表示位置が固定され、スクロールしても位置が変わらなくなります。

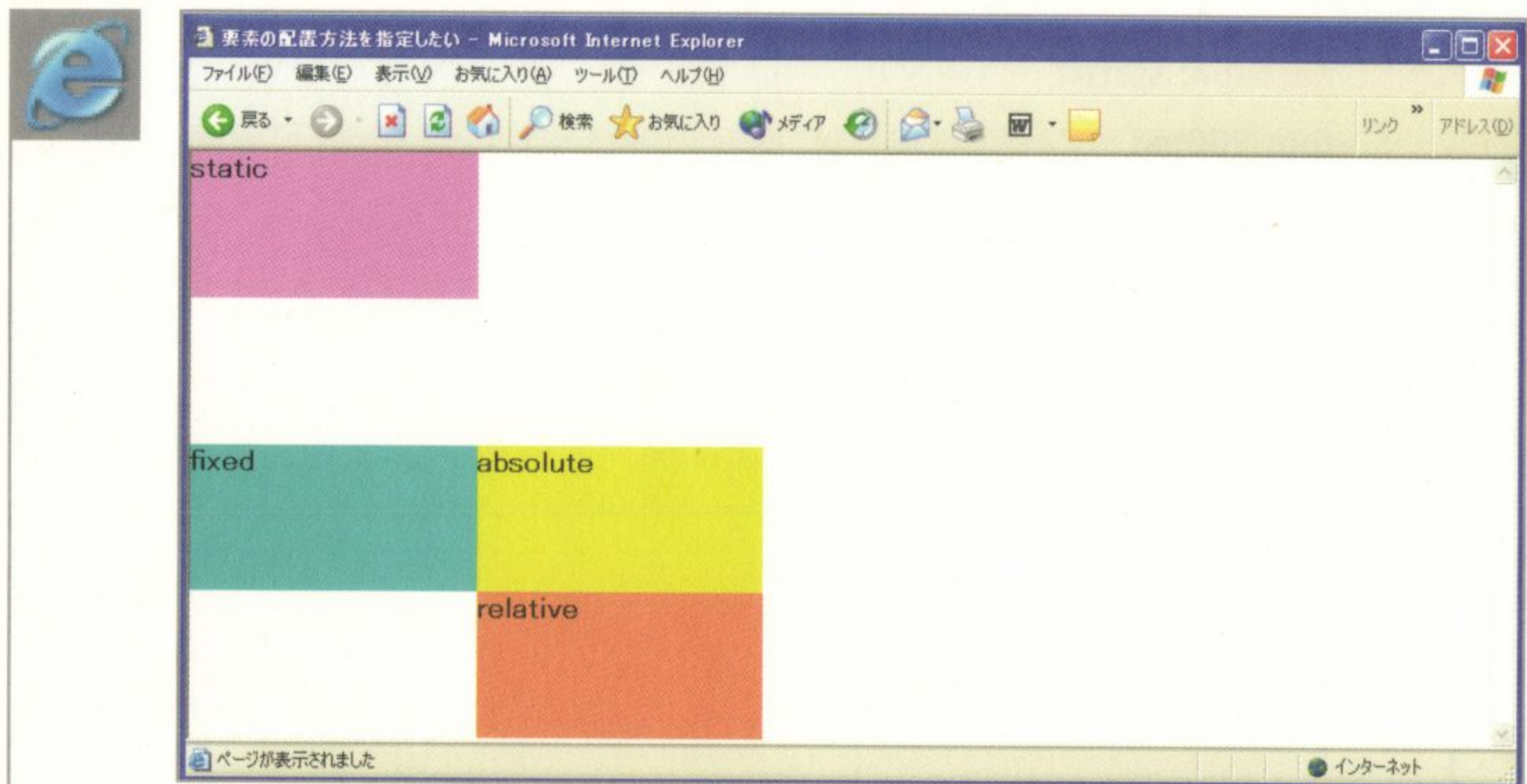
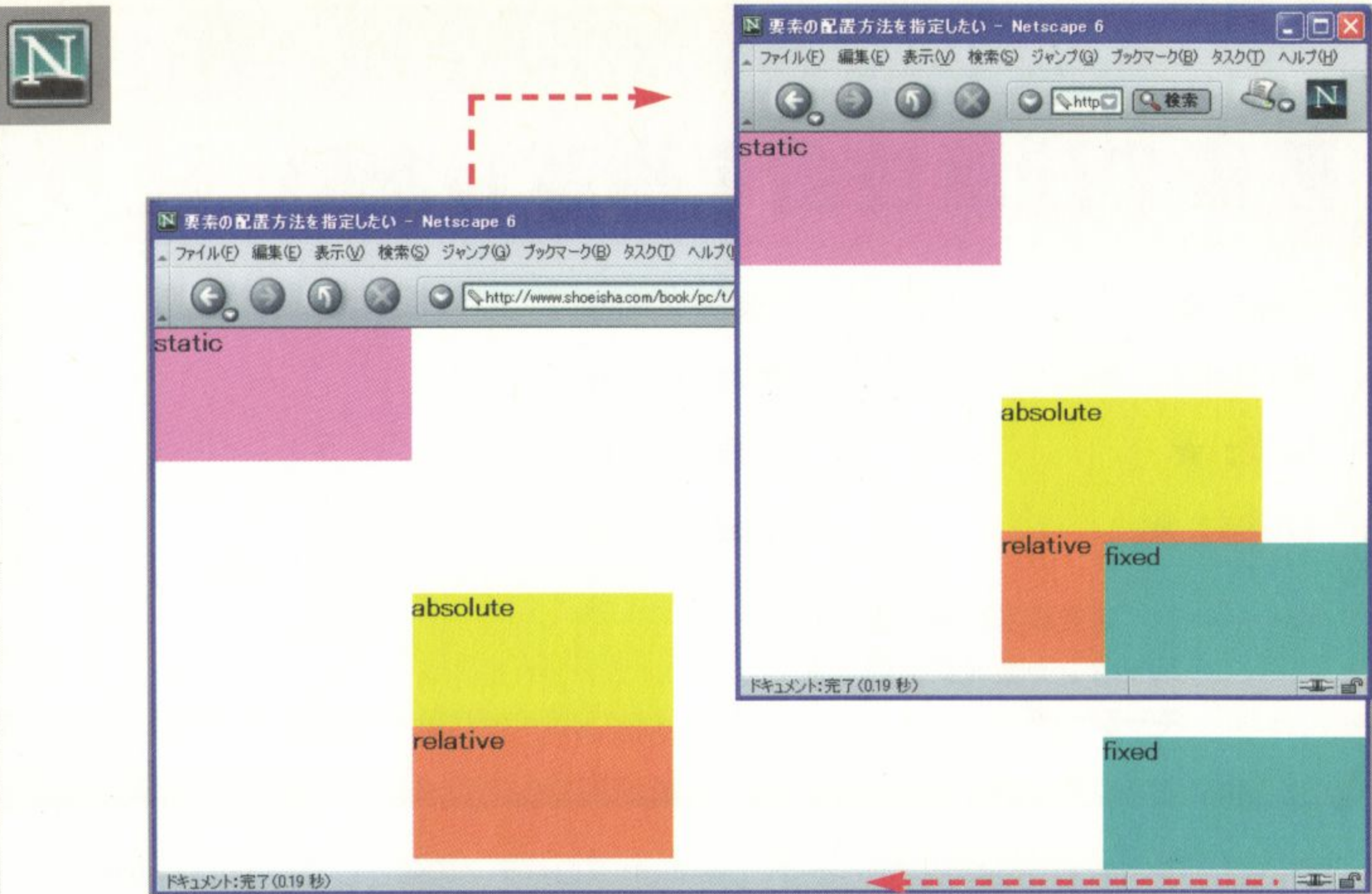
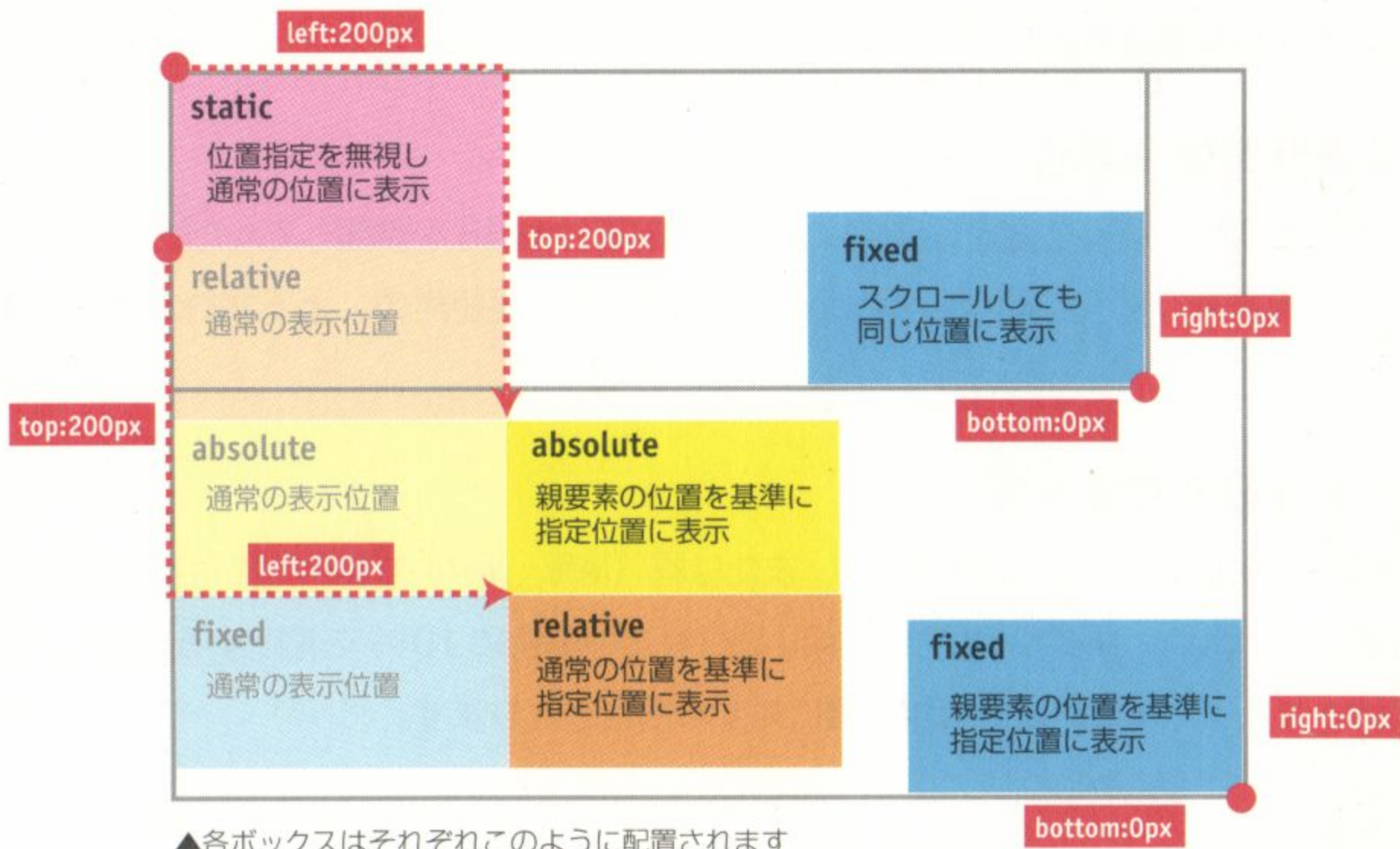
positionプロパティを単独で指定してもあまり効果はなく、通常はtop、left、right、bottomといった表示位置を指定するプロパティとともに使用します。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>表示形式を指定したい</title>
<style type="text/css">
<!--
body          { margin: 0 }
div           {
        width: 200px;
        height: 100px;
        font-weight: bold
}
.sample1      {
        position: static;
        top: 200px;
        left: 200px;
        background-color: #ff99ff
}
.sample2      {
        position: relative;
        top: 200px;
        left: 200px;
        background-color: #ff9933
}
.sample3      {
        position: absolute;
        top: 200px;
        left: 200px;
        background-color: #ffff00
}
.sample4      {
        position: fixed;
```

配置

```
        bottom: 0px;
        right: 0px;
        background-color: #00ffff
}
-->
</style>
</head>
<body>
<div class="sample1">static</div>
<div class="sample2">relative</div>
<div class="sample3">absolute</div>
<div class="sample4">fixed</div>
</body>
</html>
```

配置

▲Internet Explorerはfixedに対応していないので、fixedを指定したボックスのみ意図した表示になりません（static、absolute、relativeには対応しています）

▲Netscape 6はfixedにも対応しています。スクロールしてもfixedの位置は変わりません

▲各ボックスはそれぞれこのように配置されます

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
static	○	○	○	○	○	○	○
relative	○	○	○	○	○	○	○
absolute	○	○	○	○	○	○	○
fixed	×	×	×	×	×	×	○

※Macintosh版IE5は対応しています
※適用するセレクタによっても効果が変わります

要素の配置位置を指定したい……………………p.212
重ね合わせの順序を指定したい…………………p.222

4 POSITIONING

要素の配置位置を指定したい

top: ★ 上端からの位置
bottom: ★ 下端からの位置
left: ★ 左端からの位置
right: ★ 右端からの位置

★……サイズを表す数値＋単位
パーセントを表す数値＋%
キーワード

要素の上下左右の配置位置を指定します。positionプロパティの値としてrelative、absolute、fixedを指定しているときに利用できます。staticを指定した場合は無効になるので注意してください。

値には次のような指定方法があります。

サイズを表す数値＋単位

数値に単位をつけて位置を指定します。単位についてはp.46を参照してください。

relativeでは本来の表示位置の、absoluteとfixedでは親要素の、上下左右の端が基準になります。

パーセントを表す数値＋%

親要素の高さ（top、bottomの場合）、または幅（left、rightの場合）に対する割合で位置を指定します。たとえば、親要素の高さが400ピクセルの場合に10%を指定すると40ピクセルの位置に表示されます。relativeでも親要素に対する割合となります。

キーワード

auto 通常位置に配置（デフォルト）

autoを指定すると、配置位置は自動的に調整されます。

要素の配置位置の具体的な指定方法についてはpositionプロパティの項（p.208）を参照してください。

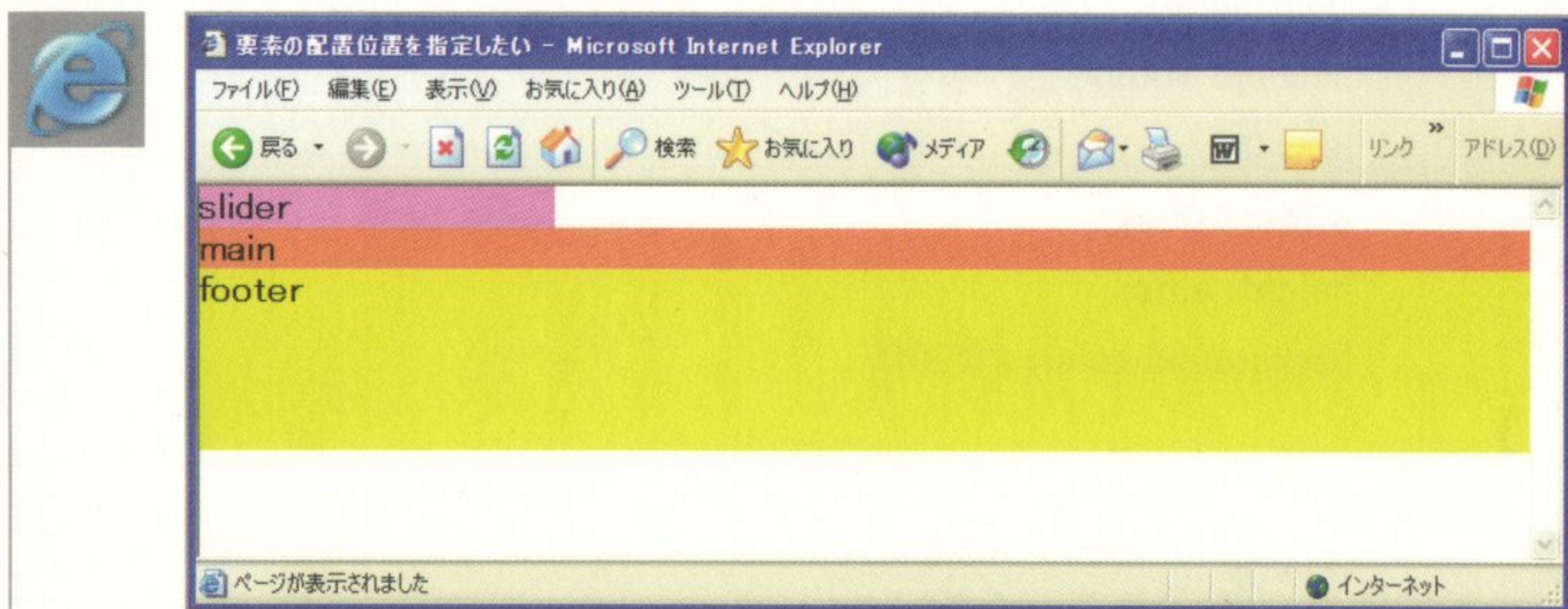
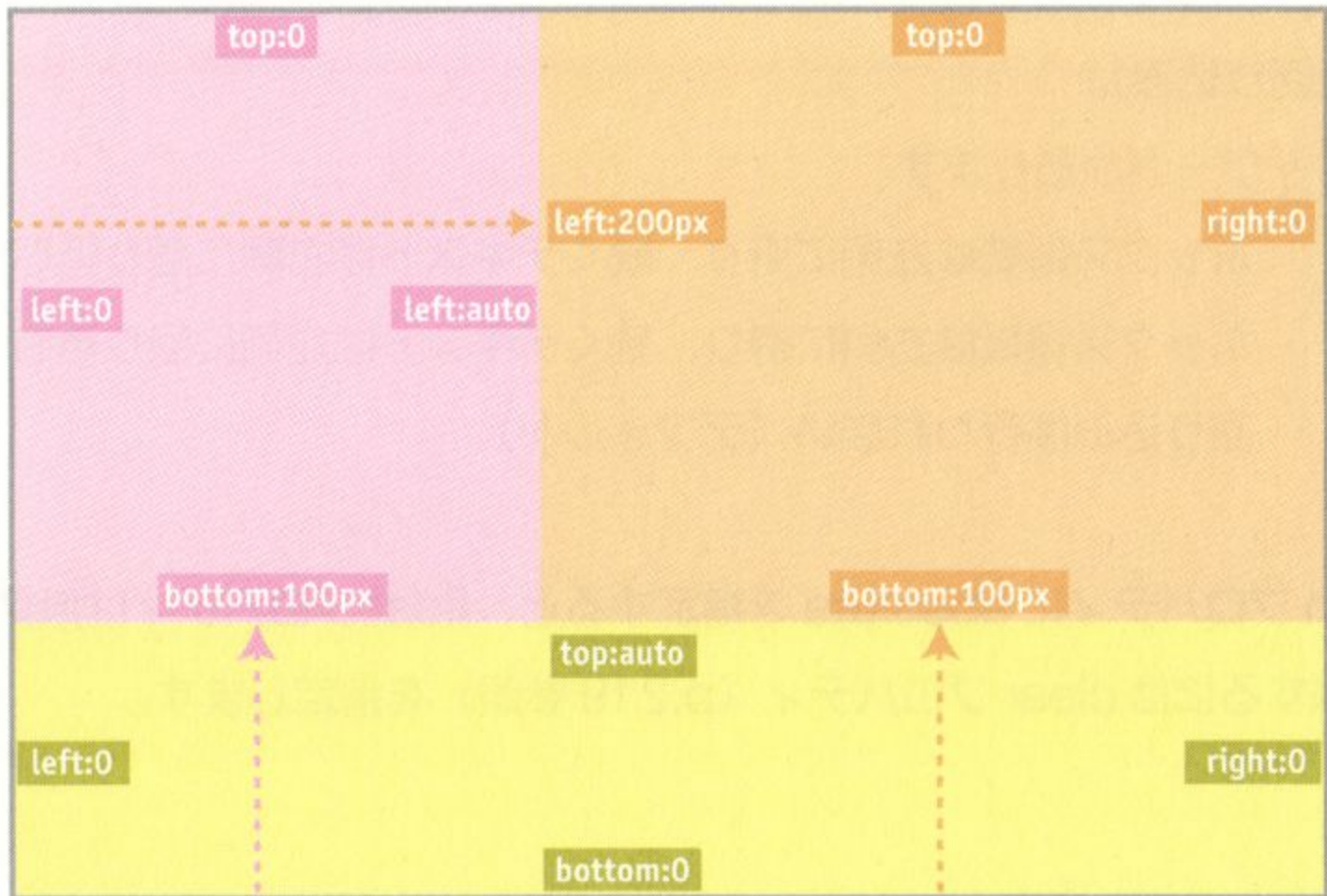
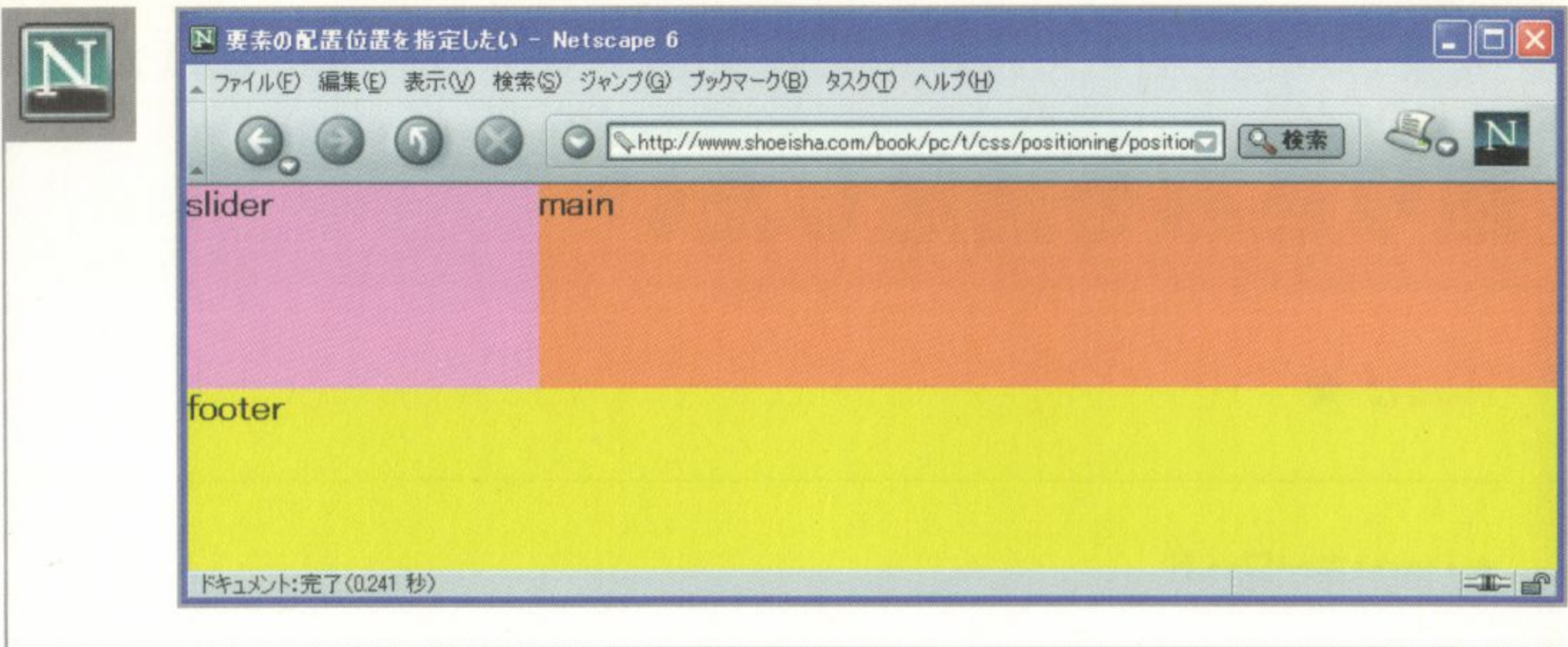
ここではposition:fixedと位置指定を利用したフレーム風のレイアウトを紹介します(Windows版のInternet Explorerはposition:fixedに対応していないのでフレーム風にはなりません)。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>要素の配置位置を指定したい</title>
<style type="text/css">
<!--
body    { margin: 0 }
div     {
        font-weight: bold;
        position: fixed
}
#slider {
        top: 0;
        right: auto;
        bottom: 100px;
        left: 0;
        width: 200px;
        height: auto;
        background-color: #ff99ff
}
#main   {
        top: 0;
        right: 0;
        bottom: 100px;
        left: 200px;
        width: auto;
        height: auto;
        background-color: #ff9933
}
#footer {
        top: auto;
        right: 0;
```

配置

```
        bottom: 0;
        left: 0;
        width: 100%;
        height: 100px;
        background-color: #ffff00
}
-->
</style>
</head>
<body>
<div id="slider">slider</div>
<div id="main">main</div>
<div id="footer">footer</div>
</body>
</html>
```

配置

▲Internet Explorerはfixedに対応していないのでこの場合は意図した表示になりません（top、right、bottom、leftには対応しています）

▲position:fixedに対応している場合、各ボックスはこのような配置関係となります

top、left	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	○	○	○
パーセント	○	○	○	○	○	○	○
auto	○	○	○	○	○	○	○

bottom、right	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	○	○	○	○	×	×	○
パーセント	○	○	○	○	×	×	○
auto	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

要素の配置方法を指定したい……………………p.208
重ね合わせの順序を指定したい…………………p.222

回り込みを指定したい

float: ★

★……キーワード

対象となるボックス領域を左、あるいは右に寄せ、その反対側に複数行のテキストを流し込むよう指定します（回り込み）。

値には以下のキーワードがあります。

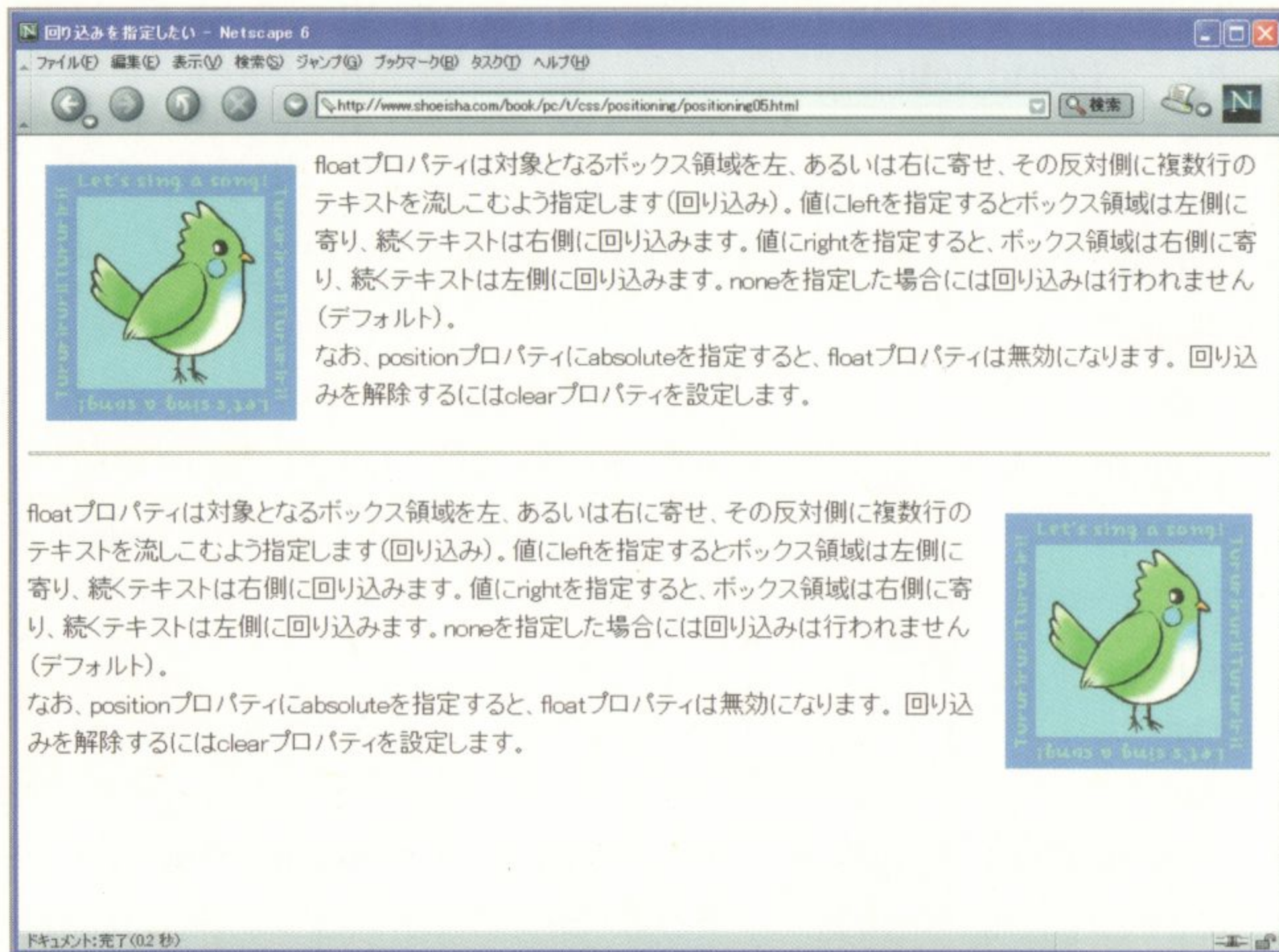
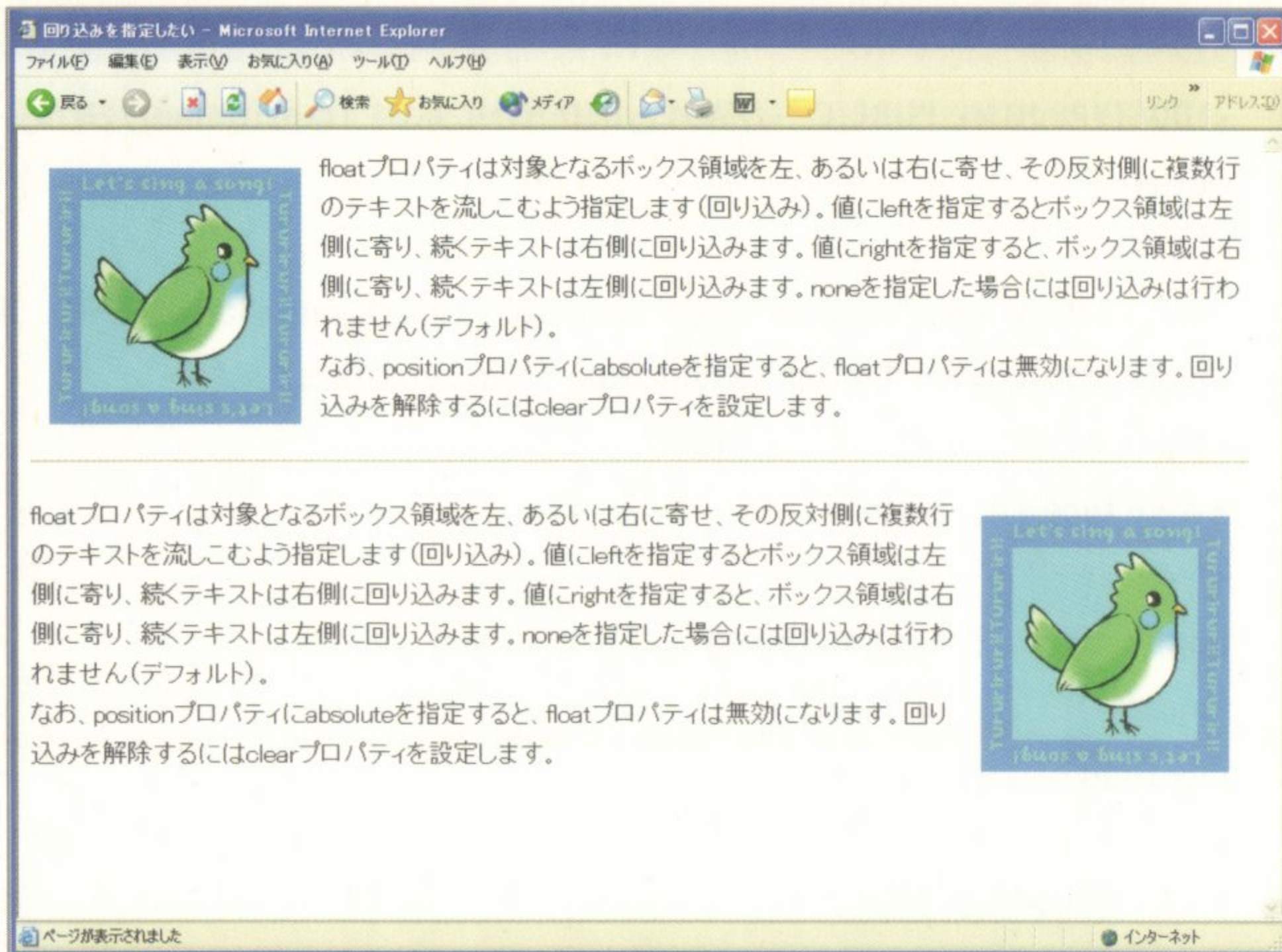
left	ボックス領域は左側に寄り、続くテキストは右側に回り込む
right	ボックス領域は右側に寄り、続くテキストは左側に回り込む
none	回り込みは行われない（デフォルト）

なお、position プロパティに absolute を指定すると、float プロパティは無効になります。

回り込みを解除するには clear プロパティ（p.219 参照）を指定します。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>回り込みを設定したい</title>
<style type="text/css">
<!--
img           { margin: 15px }
img.left      { float: left }
img.right     { float: right }
p             {
      font-size: 15pt;
      line-height: 150%
}
hr      { clear: both }
-->
</style>
</head>
<body>
<p>
<img src="birds3.jpg" class="left" width="200" height="200" alt="">
floatプロパティは……（中略）……回り込みは行われません（デフォルト）<br>
なお、positionプロパティにabsoluteを指定すると……（中略）……を設定します。
</p>
<hr>
<p>
<img src="birds3.jpg" class="right" width="200" height="200" alt="">
floatプロパティは……（中略）……回り込みは行われません（デフォルト）<br>
なお、positionプロパティにabsoluteを指定すると……（中略）……を設定します。
</p>
</body>
</html>
```

配置

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
left	○	○	○	○	○	○	○
right	○	○	○	○	○	○	○
none	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

参照 回り込みを解除したい··························p.219

6 POSITIONING

回り込みを解除したい

clear: ★

★……キーワード

floatプロパティで設定した回り込みを解除します。

値には以下のキーワードがあります。

left	左側の要素に対する回り込みを解除（float: leftの回り込みを解除）
right	右側の要素に対する回りこみを解除（float: rightの回り込みを解除）
both	回り込ませる側に関係なく、回り込みを解除
none	回り込みを解除しない（デフォルト）

配置

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>回り込みを解除したい</title>
<style type="text/css">
<!--
img          { margin: 15px }
img.left     { float: left }
img.right    { float: right }
p            {
       font-size: 15pt;
       line-height: 150%
}
br           { clear: both }
-->
```

```
</style>
</head>
<body>
<p>
<img src="birds3.jpg" class="left" width="200" height="200" alt="">
float プロパティは……（中略）……回り込みは行われません（デフォルト）。<br>
なお、position プロパティに absolute を指定すると、float プロパティは無効になります。回
り込みを解除するには clear プロパティを指定します。
</p>

<p>
<img src="birds3.jpg" class="right" width="200" height="200" alt="">
float プロパティは……（中略）……回り込みは行われません（デフォルト）。<br>
なお、position プロパティに absolute を指定すると、float プロパティは無効になります。回
り込みを解除するには clear プロパティを指定します。
</p>
</body>
</html>
```

floatプロパティは対象となるボックス領域を左、あるいは右に寄せ、その反対側に複数行のテキストを流しこむよう設定します(回り込み)。値にleftを指定するとボックス領域は左側に寄り、続くテキストは右側に回り込みます。値にrightを指定すると、ボックス領域は右側に寄り、続くテキストは左側に回り込みます。noneを指定した場合には回り込みは行われません(デフォルト)。

なお、positionプロパティにabsoluteを指定すると、floatプロパティは無効になります。回り込みを解除するにはclearプロパティを設定します。

floatプロパティは対象となるボックス領域を左、あるいは右に寄せ、その反対側に複数行のテキストを流しこむよう設定します(回り込み)。値にleftを指定するとボックス領域は左側に寄り、続くテキストは右側に回り込みます。値にrightを指定すると、ボックス領域は右側に寄り、続くテキストは左側に回り込みます。noneを指定した場合には回り込みは行われません(デフォルト)。

なお、positionプロパティにabsoluteを指定すると、floatプロパティは無効になります。回り込みを解除するにはclearプロパティを設定します。

ドキュメント:完了(0.38 秒)

配置

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
left	○	○	○	○	○	○	○
right	○	○	○	○	○	○	○
both	○	○	○	○	○	○	○
none	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

回り込みを指定したい……………………………p.216

7 POSITIONING

重ね合わせの順序を指定したい

z-index: ★

★……整数値

position や top、left、right、bottom などのプロパティによって複数の要素が重なりあった場合に、それらの重なる順序を指定します。

値には次のような指定方法があります。

配置

整数値

0 を基準にして数値が大きくなるほど前面に重なり、逆に数値が小さくなるほど背面にまわります。なお、Netscape 6.2 ではマイナスの数値を指定すると、要素自体が表示されなくなるようですので注意してください。

キーワード

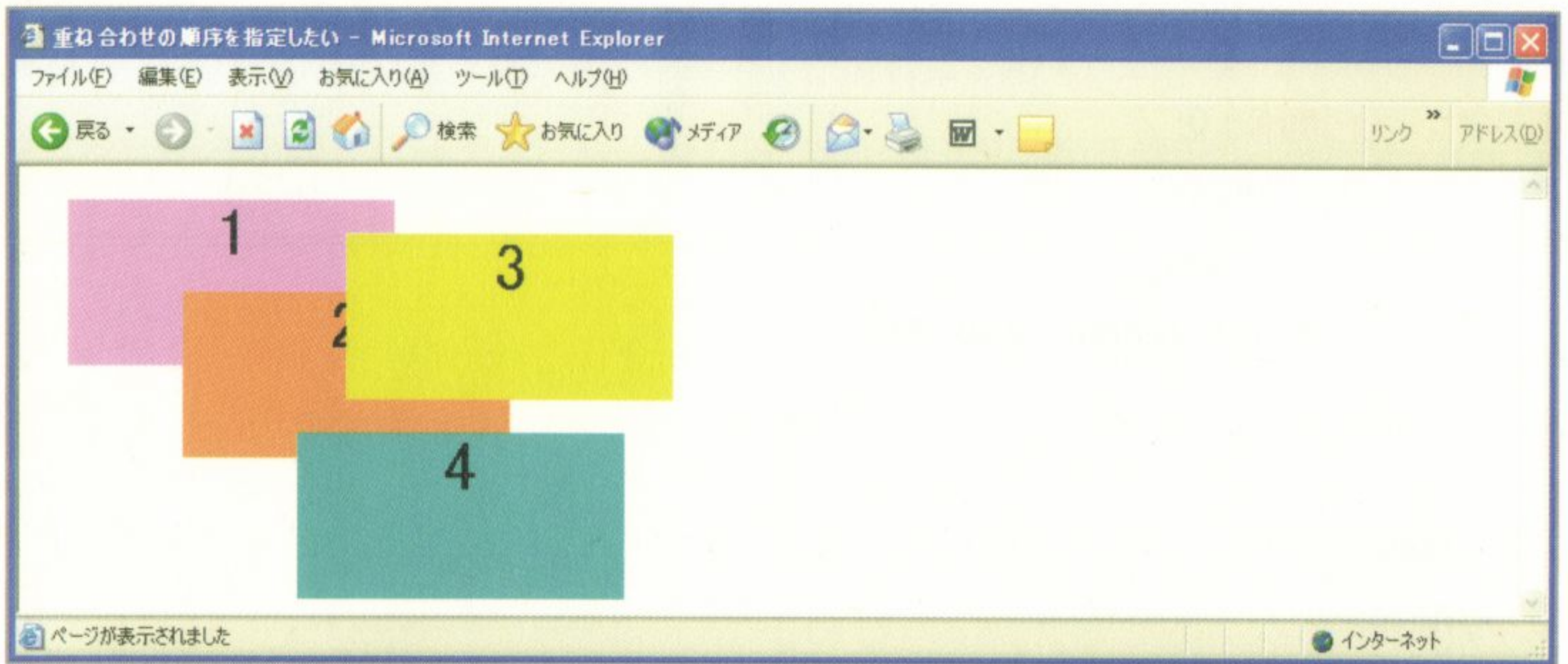
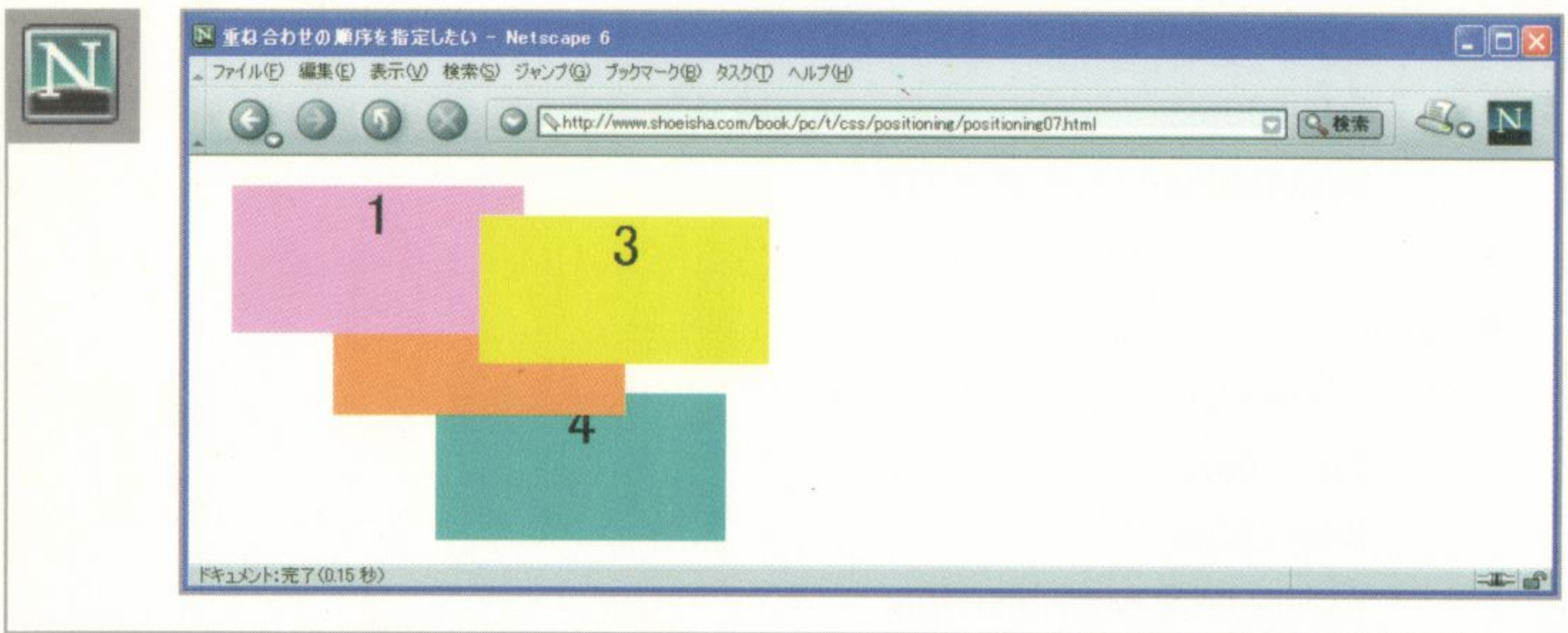
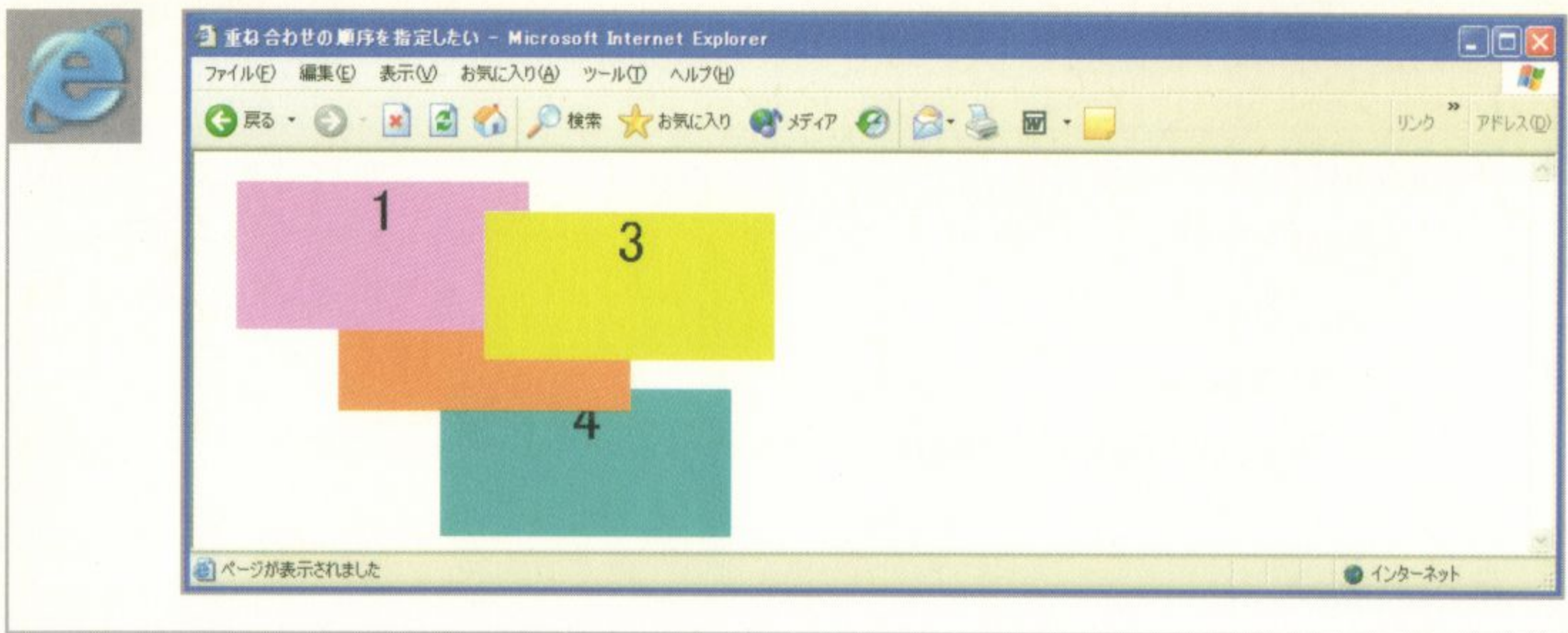
auto　　記述順（デフォルト）

通常と同じく HTML 文書での記述順に重なっていき、最後のものが最前面となります。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title> 重ね合わせの順序を指定したい </title>
<style type="text/css">
<!--
div          {
        position: absolute;
        width: 200px;
        height: 100px;
        text-align: center;
```

```
        font-size: xx-large
}
div.sample1   {
        z-index: 2;
        top: 20px;
        left: 30px;
        background-color: #ff99ff
}
div.sample2   {
        z-index: 1;
        top: 75px;
        left:100px;
        background-color: #ff9933
}
div.sample3   {
        z-index: 3;
        top: 40px;
        left: 200px;
        background-color: #ffff00
}
div.sample4   {
        z-index: 0;
        top: 160px;
        left: 170px;
        background-color: #00ffff
}
-->
</style>
</head>
<body>
<div class="sample1">1</div>
<div class="sample2">2</div>
<div class="sample3">3</div>
<div class="sample4">4</div>
</body>
</html>
```

配置

▲z-indexを指定しない場合の表示

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
数値	○	○	○	○	○	○	○
auto	○	○	○	○	○	○	○

※適用するセレクタによっても効果が変わります

要素の配置方法を指定したい…………………p.208
要素の配置位置を指定したい…………………p.212

8 POSITIONING

要素の一部を切り抜き表示したい

clip: ★

★••••••キーワード

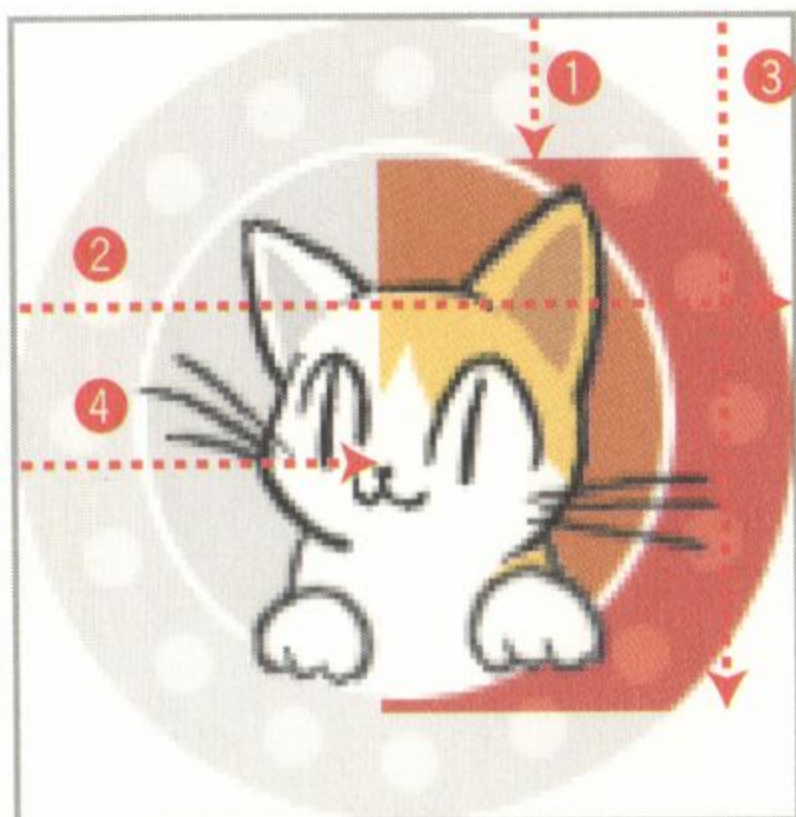
要素の見える範囲を指定し、切り抜いたように表示させます。このプロパティはpositionプロパティ（p.208参照）でabsoluteが指定された要素に適用されます。

値には以下のキーワードがあります。

auto	ボックス全体を表示（デフォルト）
rect	指定した矩形を表示

配置

表示部分の座標の指定

rectでは表示する矩形の各辺の位置を、rect(上,右,下,左)の座標のかたちで記述します。各辺の基準の位置は左上です。具体的には下の図のようになります。

clip: rectで切り抜き表示を指定するには、

clip: rect(①,②,③,④)

と表示する矩形の4辺をrectに続く括弧内に記述します。

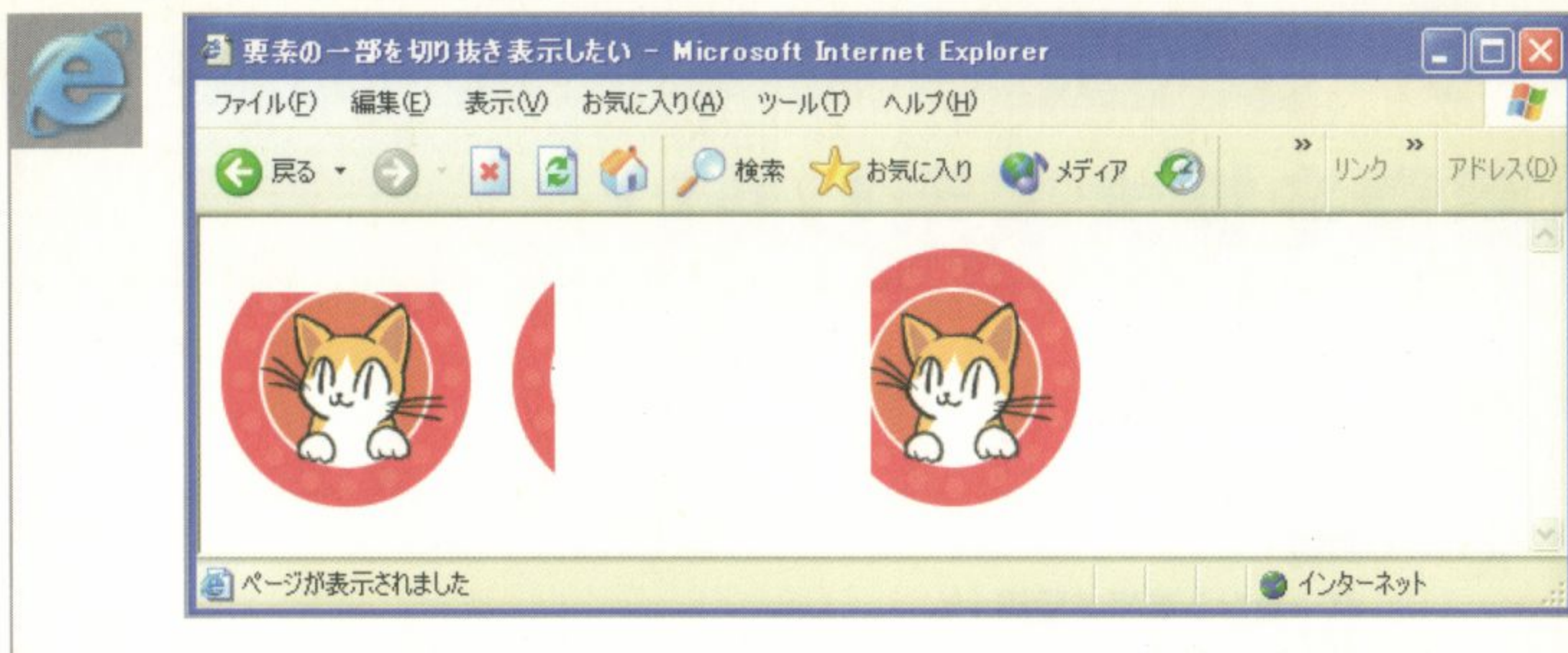
①には上辺から表示位置上端までの間隔、②には左辺から表示位置右端、③には上辺から表示位置下端、④には左辺から表示位置左端を指定してください。

座標にautoを指定すると元々の要素のボックスの辺と同じになります。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>要素の一部を切り抜き表示したい</title>
<style type="text/css">
<!--
span            { position: absolute }
span#sample1 { clip: rect(20px,auto,auto,auto) }
span#sample2 {
        clip: rect(auto,20px,auto,auto);
        left: 150px
}
span#sample3 {
        clip: rect(auto,auto,auto,20px);
        left: 300px
}
-->
</style>
</head>
<body>
<span id="sample1">
<img src="cat.gif" width="120" height="120" alt="">
</span>
<span id="sample2">
<img src="cat.gif" width="120" height="120" alt="">
</span>
<span id="sample3">
<img src="cat.gif" width="120" height="120" alt="">
</span>
</body>
</html>
```

配置

配置

CSS2におけるclipプロパティ

CSS2の仕様では、このclipプロパティは「overflowプロパティにvisible以外が指定されている要素」に適用されることになっています。また、表示する矩形の指定方法も上記の方法とは異なり、元のブロック領域の各辺からの距離を指定するよう定義されています。しかし、現在のブラウザはこの仕様に則していません。

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
rect	○	○	○	○	×	×	○
auto	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

表示・非表示を指定したい……………………p.206
要素の配置方法を指定したい…………………p.208

9 POSITIONING

要素を拡大表示したい

zoom: ★

★・・・・・・数値
パーセントを表す数値+%
キーワード

要素を拡大して表示する属性です。Internet Explorerが独自に拡張したプロパティです。
値には次のような指定方法があります。

配置

数値

本来のサイズを1としてそのサイズに対する比率を数値で指定します。2なら2倍になり、0.5なら0.5倍になります。

パーセントを表す数値+%

本来のサイズに対する比率をパーセントで指定します。100%を指定すると本来のサイズ（normal）を指定した場合と同じ結果になります。

キーワード

normal	本来のサイズ（デフォルト）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>要素を拡大表示したい</title>
<style type="text/css">
<!--
.sample1      { zoom: 150% }
.sample2      { zoom: 0.5 }
```

```
div              { margin: 30px }
-->
</style>
</head>
<body>
<div>
<span> ピヨ </span><span class="sample1"> ピヨ </span>
<span class="sample2"> ピヨ </span>
</div>
<div>
<img src="cat.gif" width="120" height="120" alt="">
<img src="cat.gif" class="sample1" width="120" height="120" alt="">
<img src="cat.gif" class="sample2" width="120" height="120" alt="">
</div>
<form>
   <input type="button" value="クリック！">
   <input type="button" value="クリック！" class="sample1">
   <input type="button" value="クリック！" class="sample2">
</form>
</body>
</html>
```


▲Netscapeは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
数値	×	×	○	○	×	×	×
パーセント	×	×	○	○	×	×	×

※適用するセレクタによっても効果が変わります

10 POSITIONING

内容があふれる場合の処理方法を指定したい

overflow: ★

★……キーワード

heightやwidthでボックス領域の幅や高さを指定すると、コンテンツの量によってはボックス領域内に収まりきらないことがあります。このようにコンテンツがあふれる場合の処理方法を指定します。

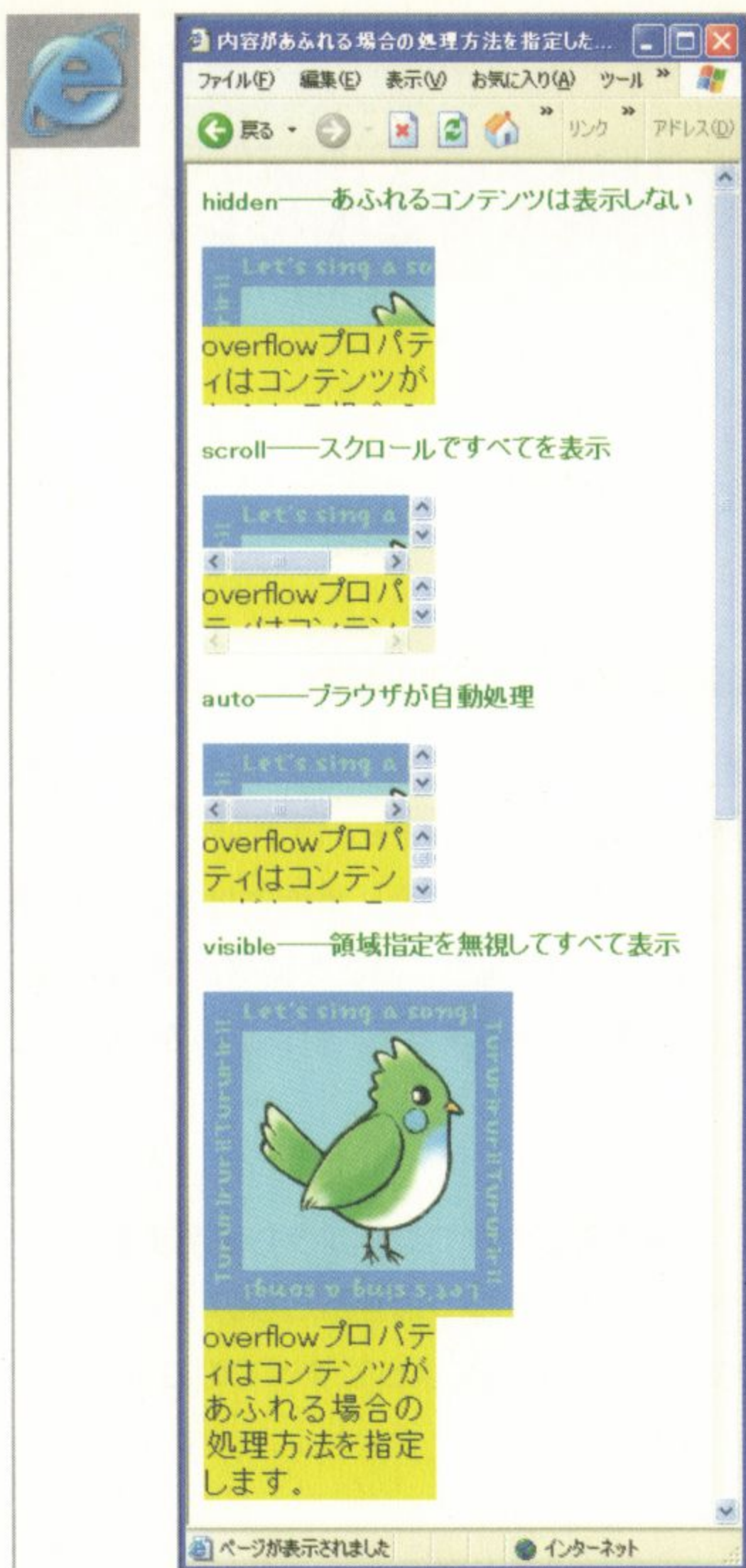
値には以下のキーワードがあります。

visible	領域指定を無視して高さや幅を調整し、収まりきらないコンテンツも表示する（デフォルト）
hidden	領域指定にしたがって、収まりきらないコンテンツは表示しない
scroll	縦横にスクロールバーをつけて、スクロールによってすべてのコンテンツを表示
auto	ブラウザが自動的に処理

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>内容があふれる場合の処理方法を指定したい</title>
<style type="text/css">
<!--
p               {
        color: #008000;
        font-size: 12pt;
        font-weight: bold
}
div             {
        height: 50px;
        width: 150px;
        background-color: #ffff00
}
div.sample1     { overflow: hidden }
div.sample2     { overflow: scroll }
div.sample3     { overflow: auto }
div.sample4     { overflow: visible }
-->
</style>
</head>
<body>

<p>hidden――あふれるコンテンツは表示しない</p>
<div class="sample1">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample1">
overflowプロパティはコンテンツがあふれる場合の処理方法を指定します。
</div>
```

配置

```
<p>scroll――スクロールですべてを表示</p>
<div class="sample2">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample2">
overflow プロパティはコンテンツがあふれる場合の処理方法を指定します。
</div>

<p>auto――ブラウザが自動処理</p>
<div class="sample3">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample3">
overflow プロパティはコンテンツがあふれる場合の処理方法を指定します。
</div>

<p>visible――領域指定を無視してすべて表示</p>
<div class="sample4">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample4">
overflow プロパティはコンテンツがあふれる場合の処理方法を指定します。
</div>

</body>
</html>
```

▲Netscapeでは、visibleを指定した要素があふれて領域指定を無視しても、他の要素は本来の位置に表示されるため、一部の要素が重なることがあります

配置

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
visible	×	○	○	○	×	×	○
scroll	×	○	○	○	×	×	○
hidden	×	○	○	○	×	×	○
auto	×	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

内容領域の幅と高さを指定したい……………p.198
内容があふれる場合の横方向の処理方法を指定したい……p.234
内容があふれる場合の縦方向の処理方法を指定したい……p.238
スクロールバーの色を指定したい……………p.320

内容があふれる場合の横方向の処理方法を指定したい

overflow-x: ★

★……キーワード

コンテンツがあふれる場合の横方向の処理方法を指定します。overflowプロパティ（p.230参照）のバリエーションとしてInternet Explorerが独自に拡張したプロパティです。

ただし、現状のInternet Explorerではコンテンツがあふれる場合、標準では高さの制限（height）を無視します。そのため、コンテンツがテキストである場合、高さを調整することですべてが表示されてしまうので、overflow-xを指定しても意味はありません。サンプルではテキストに改行不可を指定して、効果を表現しています。

値には以下のキーワードがあります。

visible	領域指定を無視して幅を調整し、収まりきらないコンテンツも表示する
hidden	領域指定にしたがって、収まりきらないコンテンツは表示しない
scroll	横方向のスクロールバーをつけて、スクロールによってすべてのコンテンツを表示
auto	ブラウザが自動的に処理

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>内容があふれる場合の横方向の処理方法を指定したい</title>
<style type="text/css">
<!--
span          { color: #ff0000 }
div           {
       font-size:12pt;
       height: 50px;
       width: 150px;
       white-space: nowrap;
       background-color: #ffff00
}
div.sample1   { overflow-x: visible }
div.sample2   { overflow-x: hidden }
div.sample3   { overflow-x: scroll }
div.sample4   { overflow-x: auto }
-->
</style>
</head>
<body>

<div>指定サイズ</div><br>

<div class="sample1">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample1">
<span>overflow-x: visible</span>――領域指定を無視してすべて表示
</div>
<br>
```

配置

```
<div class="sample2">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample2">
<span>overflow-x: hidden</span> ――あふれるコンテンツは表示しない
</div>
<br>

<div class="sample3">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample3">
<span>overflow-x: scroll</span> ――スクロールですべてを表示
</div>
<br>

<div class="sample4">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample4">
<span>overflow-x: auto</span> ――ブラウザが自動処理
</div>

</body>
</html>
```

配置

▲width: 150pxの領域指定に対して値に応じた処理を行います

▲Netscapeは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
visible	×	○	○	○	×	×	×
scroll	×	○	○	○	×	×	×
hidden	×	○	○	○	×	×	×
auto	×	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります
※Macintosh版IE5は対応していません

参照
内容領域の幅と高さを指定したい……………p.198
内容があふれる場合の処理方法を指定したい……p.230
内容があふれる場合の縦方向の処理方法を指定したい……p.238
スクロールバーの色を指定したい……………p.320

12 POSITIONING

内容があふれる場合の縦方向の処理方法を指定したい

overflow-y: ★

★……キーワード

コンテンツがあふれる場合の縦方向の処理方法を指定します。overflowプロパティ（p.230参照）のバリエーションとしてInternet Explorerが独自に拡張したプロパティです。

overflow-yはoverflow-xとは異なり、横方向の制限を無視することはないので（画像の場合は無視）、コンテンツがテキストであっても値に応じた処理を行います。

値には以下のキーワードがあります。

visible	領域指定を無視して幅を調整し、収まりきらないコンテンツも表示する
hidden	領域指定にしたがって、収まりきらないコンテンツは表示しない
scroll	縦方向のスクロールバーをつけて、スクロールによってすべてのコンテンツを表示
auto	ブラウザが自動的に処理

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>内容があふれる場合の縦方向の処理方法を指定したい</title>
<stile type="text/css">
<!--
span            { color: #ff0000 }
div             {
        font-size:16pt;
        height: 50px;
        width: 150px;
        background-color: #ffff00
}
div.sample1     { overflow-y: visible }
div.sample2     { overflow-y: hidden }
div.sample3     { overflow-y: scroll }
div.sample4     { overflow-y: auto }
-->
</style>
</head>
<body>

<div>指定サイズ</div><br>

<div class="sample1">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample1">
<span>overflow-y: visible</span>――領域指定を無視してすべて表示
</div>
<br>
```

配置

```
<div class="sample2">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample2">
<span>overflow-y: hidden</span> ──あふれるコンテンツは表示しない
</div>
<br>

<div class="sample3">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample3">
<span>overflow-y: scroll</span> ──スクロールですべてを表示
</div>
<br>

<div class="sample4">
<img src="birds3.jpg" width="200" height="200" alt="">
</div>
<div class="sample4">
<span>overflow-y: auto</span> ──ブラウザが自動処理
</div>

</body>
</html>
```

配置

▲height: 50pxの領域指定に対して値に応じた処理を行います

▲Netscapeは対応していません

配置

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
visible	×	○	○	○	×	×	×
scroll	×	○	○	○	×	×	×
hidden	×	○	○	○	×	×	×
auto	×	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります
※Macintosh版IE5は対応していません

内容領域の幅と高さを指定したい……………p.198
内容があふれる場合の処理方法を指定したい……p.230
内容があふれる場合の横方向の処理方法を指定したい……p.234
スクロールバーの色を指定したい……………p.320

1 LIST

リストのマークを指定したい

list-style-type: ★

★……キーワード

リストのマークの種類を指定します。

値には以下のキーワードがあります。ただし、list-style-image（p.246参照）で画像が指定されている場合はそちらの設定を優先します。

none	マークなし
disc	黒丸（デフォルト）
circle	白丸
square	四角
decimal	10進数（1から始まる数字）
decimal-leading-zero	0を頭につけた10進数（01,02,03,…98,99）
lower-roman	小文字ローマ数字
upper-roman	大文字ローマ数字
lower-greek	小文字ギリシア文字
lower-alpha	小文字アルファベット
lower-latin	小文字アルファベット
upper-alpha	大文字アルファベット
upper-latin	大文字アルファベット
hebrew	ヘブライ数字
armenian	アルメニア数字
georgian	グルジア数字
cjk-ideographic	漢数字（一,二,三）
hiragana	ひらがな（あいうえお順）
katakana	カタカナ（アイウエオ順）
hiragana-iroha	ひらがなのいろは（いろは順）
katakana-iroha	カタカナのイロハ（イロハ順）

none
none
none

▲none

• disc
• disc
• disc

▲disc

○ circle
○ circle
○ circle

▲circle

■ square
■ square
■ square

▲square

1. decimal
2. decimal
3. decimal

▲decimal

01. decimal-leading-zero
02. decimal-leading-zero
03. decimal-leading-zero

▲decimal-leading-zero

i. lower-roman
ii. lower-roman
iii. lower-roman

▲lower-roman

I. upper-roman
II. upper-roman
III. upper-roman

▲upper-roman

α. lower-greek
β. lower-greek
γ. lower-greek

▲lower-greek

a. lower-alpha
b. lower-alpha
c. lower-alpha

▲lower-alpha

a. lower-latin
b. lower-latin
c. lower-latin

▲lower-latin

A. upper-alpha
B. upper-alpha
C. upper-alpha

▲upper-alpha

A. upper-latin
B. upper-latin
C. upper-latin

▲upper-latin

א. hebrew
ב. hebrew
ג. hebrew

▲hebrew

Ա. armenian
Բ. armenian
Գ. armenian

▲armenian

ა. georgian
ბ. georgian
გ. georgian

▲georgian

一. cjk-idasseographic
二. cjk-ideographic
三. cjk-ideographic

▲cjk-ideographic

あ. hiragana
い. hiragana
う. hiragana

▲hiragana

ア. katakana
イ. katakana
ウ. katakana

▲katakana

い. hiragana-iroha
ろ. hiragana-iroha
は. hiragana-iroha

▲hiragana-iroha

イ. katakana-iroha
ロ. katakana-iroha
ハ. katakana-iroha

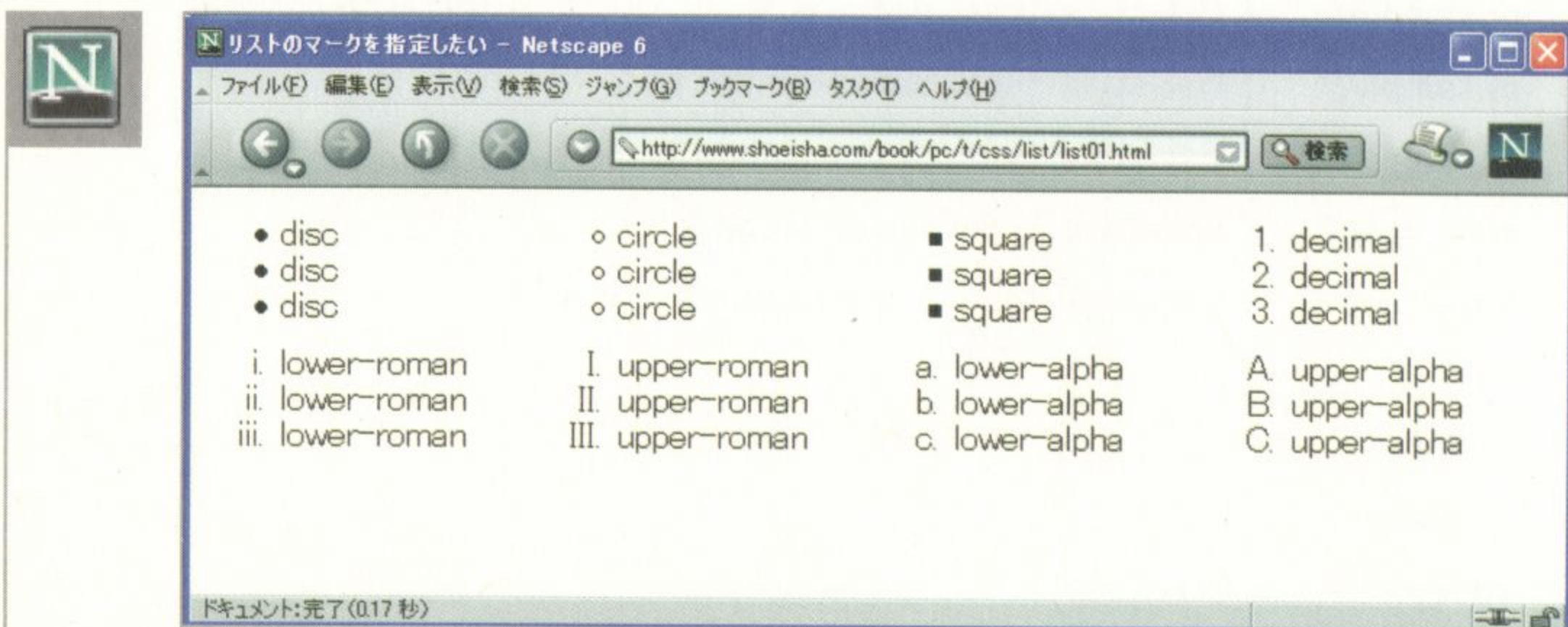
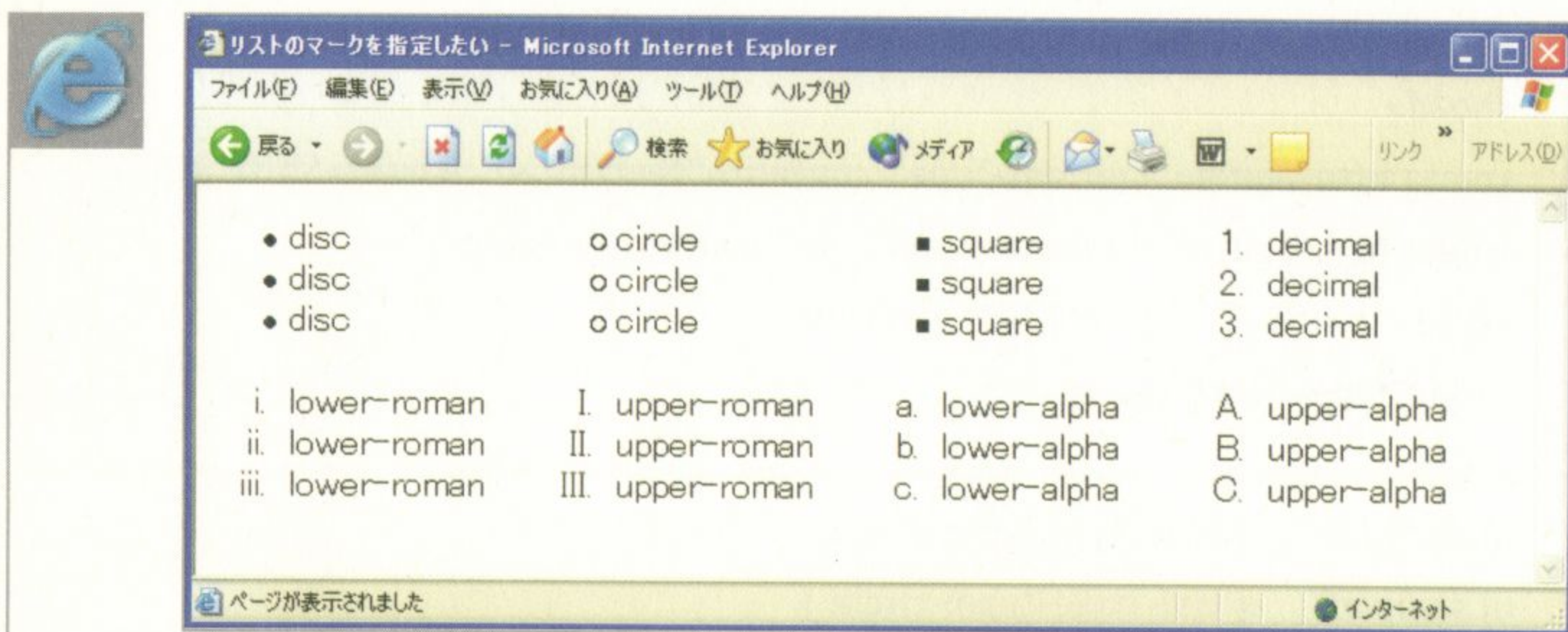
▲katakana-iroha

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>リストのマークを指定したい</title>
<style type="text/css">
<!--
ul.sample1    { list-style-type: disc }
ul.sample2    { list-style-type: circle }
ul.sample3    { list-style-type: square }
ol.sample4    { list-style-type: decimal }
ol.sample5    { list-style-type: lower-roman }
ol.sample6    { list-style-type: upper-roman }
ol.sample7    { list-style-type: lower-alpha }
olsample8     { list-style-type: upper-alpha }
-->
</style>
</head>
<body>
```

```
<ul class="sample1"><li>disc</li><li>disc</li><li>disc</li></ul>
<ul class="sample2"><li>circle</li><li>circle</li><li>circle</li></ul>
<ul class="sample3"><li>square</li><li>square</li><li>square</li></ul>
<ol class="sample4"><li>decimal</li><li>decimal</li><li>decimal</li></ol>
<ol class="sample5"><li>lower-roman </li><li>lower-roman</li>
<li>lower-roman</li></ol>
<ol class="sample6"><li>upper-roman</li><li>upper-roman</li>
<li>upper-roman</li></ol>
<ol class="sample7"><li>lower-alpha</li><li>lower-alpha</li>
<li>lower-alpha</li></ol>
<ol class="sample8"><li>upper-alpha</li><li>upper-alpha</li>
<li>upper-alpha</li></ol>
</body>
</html>
```

※レイアウトの都合上、実際にはtableタグを組んでいますがソースは割愛します

リストマークを指定するHTMLタグをCSSに改める

HTMLタグでリストのマークを指定するには、次のように<ul>、<ol>または<li>タグにtype属性を指定します。

```
<ul type="disc">～</ul>
<ul type="square">～</ul>
<ul type="circle">～</ul>
```

```
<ol type="i">～</ol>
<ol type="1">～</ol>
<ol type="a">～</ol>
<ol type="A">～</ol>
```

type属性によってマークを変更する方法はDeprecated（推奨しない）とされており、リストのマークはスタイルシートで指定することが推奨されています。同様の効果をスタイルシートで表現すると以下のようになります。

```
ul { list-style-type: disc }
ul { list-style-type: square }
ul { list-style-type: circle }
```

```
ol { list-style-type: lower-roman }
ol { list-style-type: decimal }
ol { list-style-type: lower-alpha }
ol { list-style-type: upper-roman }
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
disc	○	○	○	○	○	○	○
circle	○	○	○	○	○	○	○
square	○	○	○	○	○	○	○
decimal	○	○	○	○	○	○	○
decimal-leading-zero	×	×	×	×	×	×	○
lower-roman	○	○	○	○	○	○	○
upper-roman	○	○	○	○	○	○	○
lower-greek	×	×	×	×	×	×	○
lower-alpha	○	○	○	○	○	○	○
lower-latin	×	×	×	×	×	×	○
upper-alpha	○	○	○	○	○	○	○
upper-latin	×	×	×	×	×	×	○
hebrew	×	×	×	×	×	×	○
armenian	×	×	×	×	×	×	○
georgian	×	×	×	×	×	×	○
cjk-ideographic	×	×	×	×	×	×	○
hiragana	×	×	×	×	×	×	○
katakana	×	×	×	×	×	×	○
hiragana-iroha	×	×	×	×	×	×	○
katakana-iroha	×	×	×	×	×	×	○
none	○	○	○	○	×	○	○

※適用するセレクタによっても効果が変わります
※Macintosh版N6.2はhebrew、armenian、georgianには対応していません

リストのマークに画像を使用したい……………p.246
リストのマークの配置を指定したい……………p.248
リストのマークを一括して指定したい…………p.250

2 LIST

リストのマークに画像を使用したい

list-style-image: ★

★……URL
キーワード

リストのマークとして表示させる画像を指定します。

値には次のような指定方法があります。なお、list-style-type（p.242参照）と同時に設定されている場合には、list-style-imageの指定が優先され、指定された画像ファイルが見つからなかった場合にlist-style-typeの指定が適用されます。

URL

list-style-image: url("☆")　　☆……画像ファイルのURL

表示させる画像ファイルのURLを上記の形式で指定します。

HTML文書から外部のスタイルシートを読み込む場合はHTML文書からの相対URLではなく、スタイルファイルからの相対URLで指定しなくてはなりません（p.50参照）。

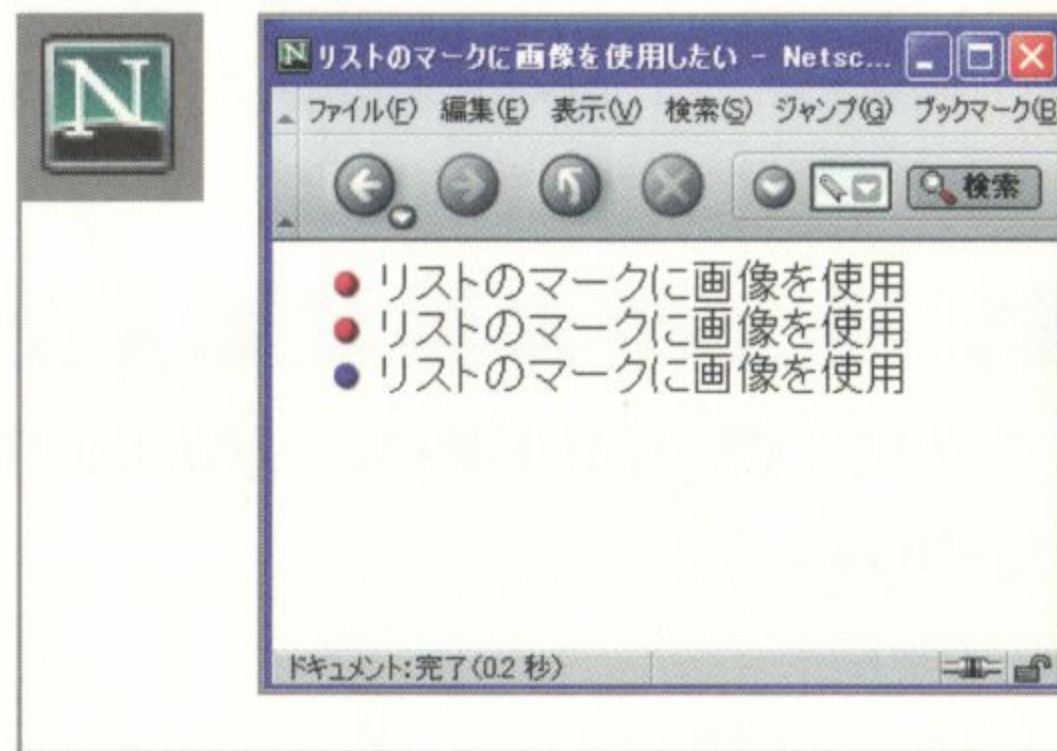
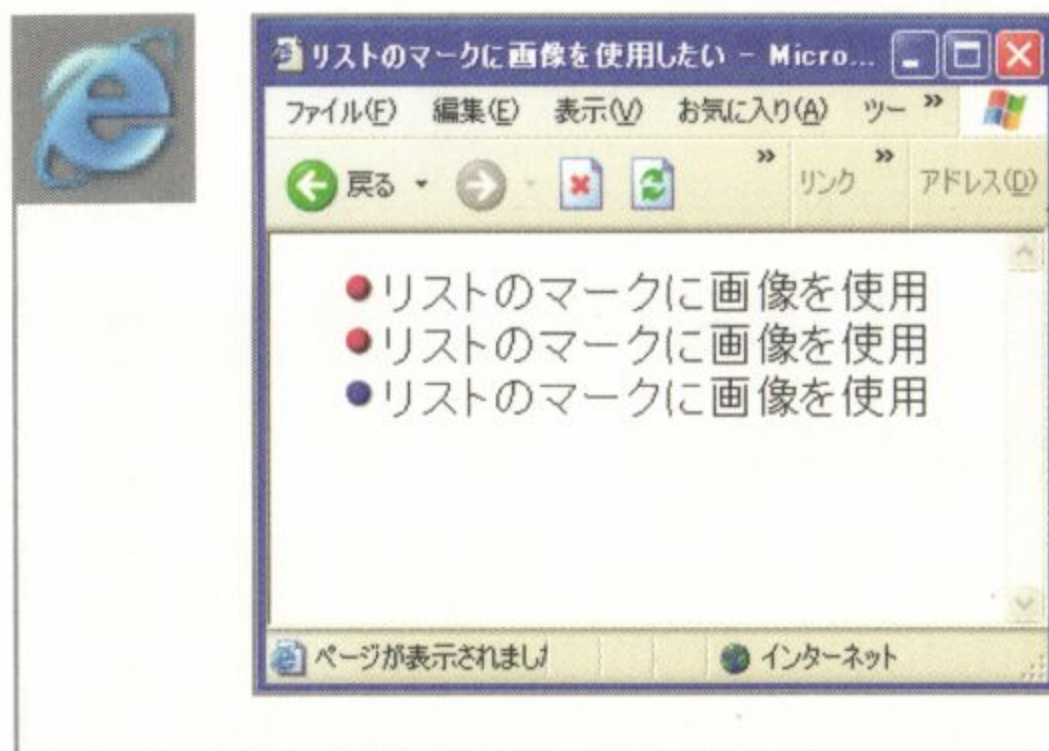
キーワード

none　画像を表示しない（デフォルト）

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>リストのマークに画像を使用したい</title>
<style type="text/css">
<!--
ul          { list-style-image: url("ball_red.gif") }
li.blue     { list-style-image: url("ball_blue.gif") }
```

```
-->
</style>
</head>
<body>
<ul>
  <li> リストのマークに画像を使用 </li>
  <li> リストのマークに画像を使用 </li>
  <li class="blue"> リストのマークに画像を使用 </li>
</ul>
</body>
</html>
```

リスト

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
URL	○	○	○	○	×	×	○
none	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

リストのマークを指定したい……………………p.242
リストのマークの配置を指定したい……………p.248
リストのマークを一括して指定したい…………p.250

3 LIST

リストのマークの配置を指定したい

list-style-position: ★

★••••••キーワード

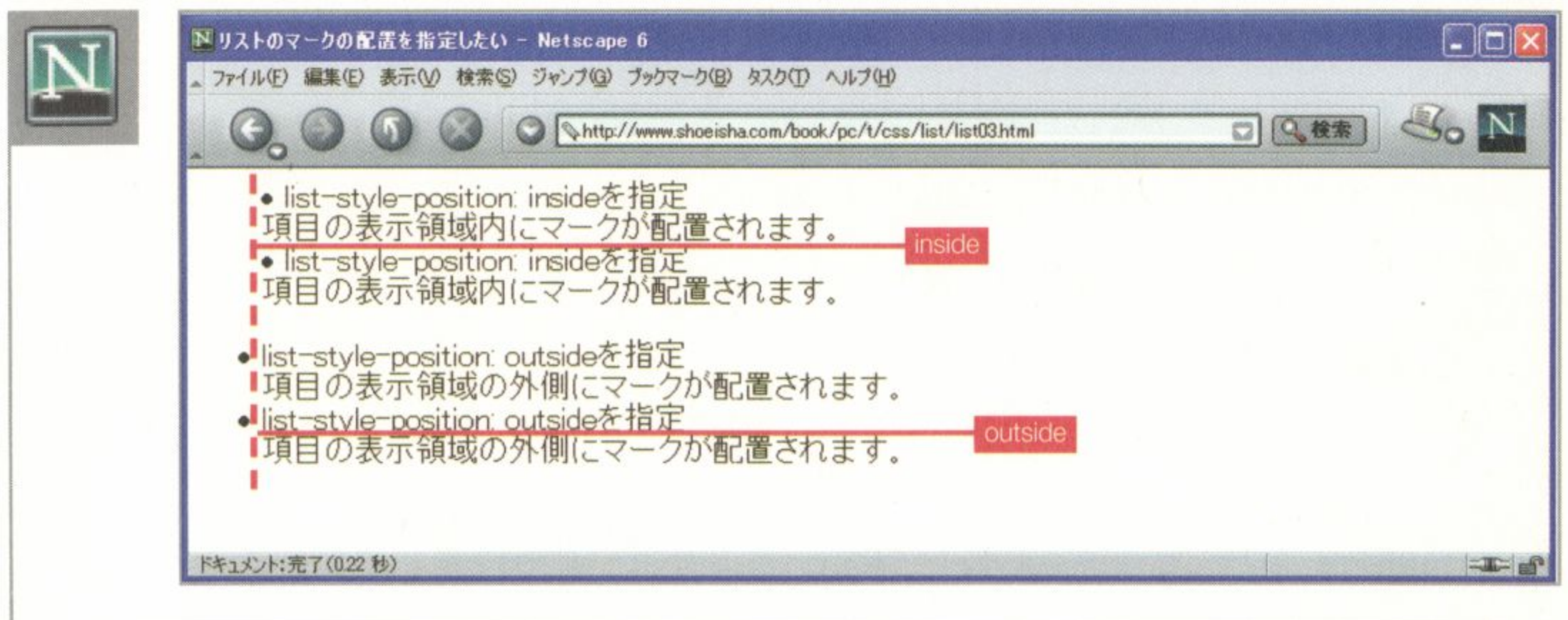
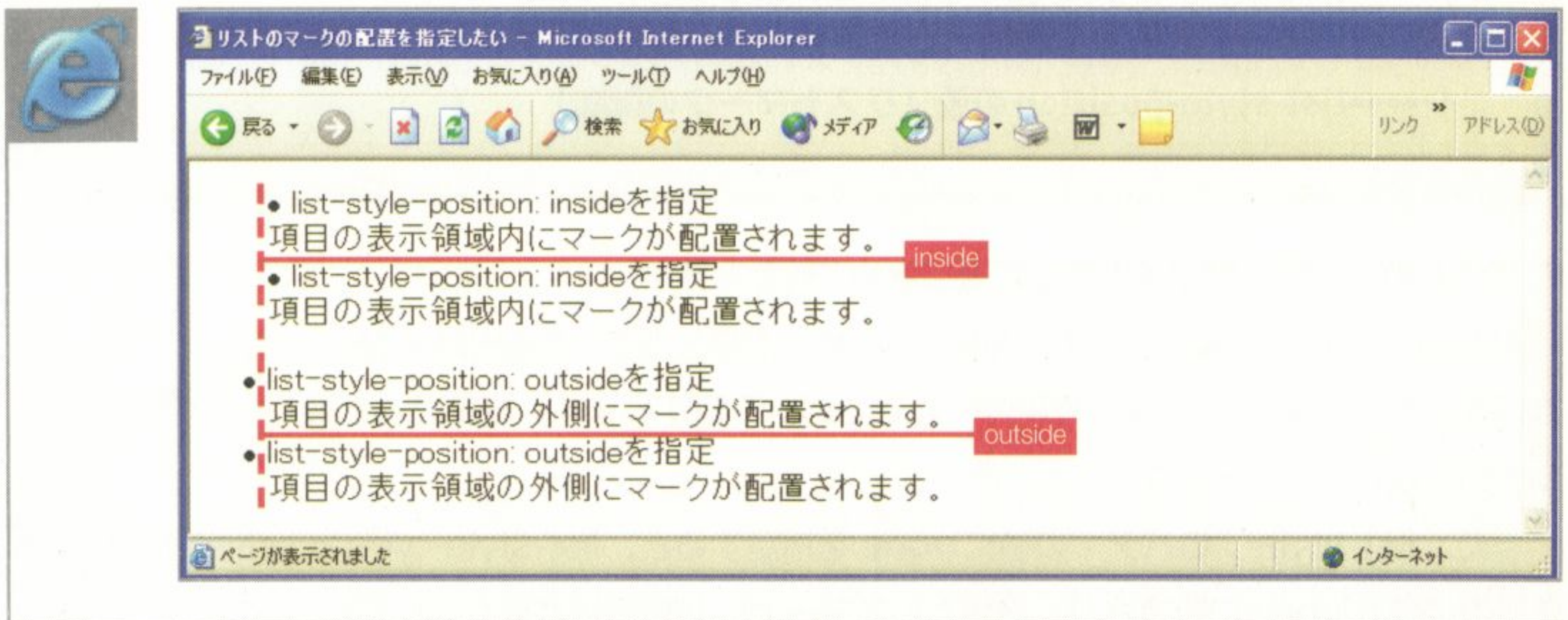
リストのマークをリストの項目の表示領域の外側に置くか、内側に置くかを指定します。
値には以下のキーワードがあります。

outside	マークを項目の外側に配置（デフォルト）
inside	マークを項目の内側に配置

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>リストのマークの配置を指定したい</title>
<style type="text/css">
<!--
ul#in   { list-style-position: inside }
ul#out  { list-style-position: outside }
-->
</style>
</head>
<body>
<ul id="in">
  <li>list-style-position: insideを指定<br>項目の表示領域内にマークが配置されます。</li>
  <li>list-style-position: insideを指定<br>項目の表示領域内にマークが配置されます。</li>
</ul>
```

リスト

```
<ul id="out">
    <li>list-style-position: outside を指定 <br> 項目の表示領域の外側にマークが配置されます。</li>
    <li>list-style-position: outside を指定 <br> 項目の表示領域の外側にマークが配置されます。</li>
</ul>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
inside	○	○	○	○	×	×	○
outside	○	○	○	○	×	×	○

※適用するセレクタによっても効果が変わります

リストのマークを指定したい……………………p.242　リストのマークを一括して指定したい…………p.250
リストのマークに画像を使用したい……………p.246

4 LIST

リストのマークを一括して指定したい

list-style: ★ ☆ △

★••••••list-style-type の値（リストマークの種類）
☆••••••list-style-image の値（リストマークの画像）
△••••••list-style-position の値（リストマークの配置）

リストのマークに関する各種指定をまとめて設定します。

list-style-type（種類）、list-style-image（画像）、list-style-position（配置）のうち必要な値を任意の順番で並べ、半角スペースで区切って指定します。省略された値についてはデフォルトの設定が適用されます。

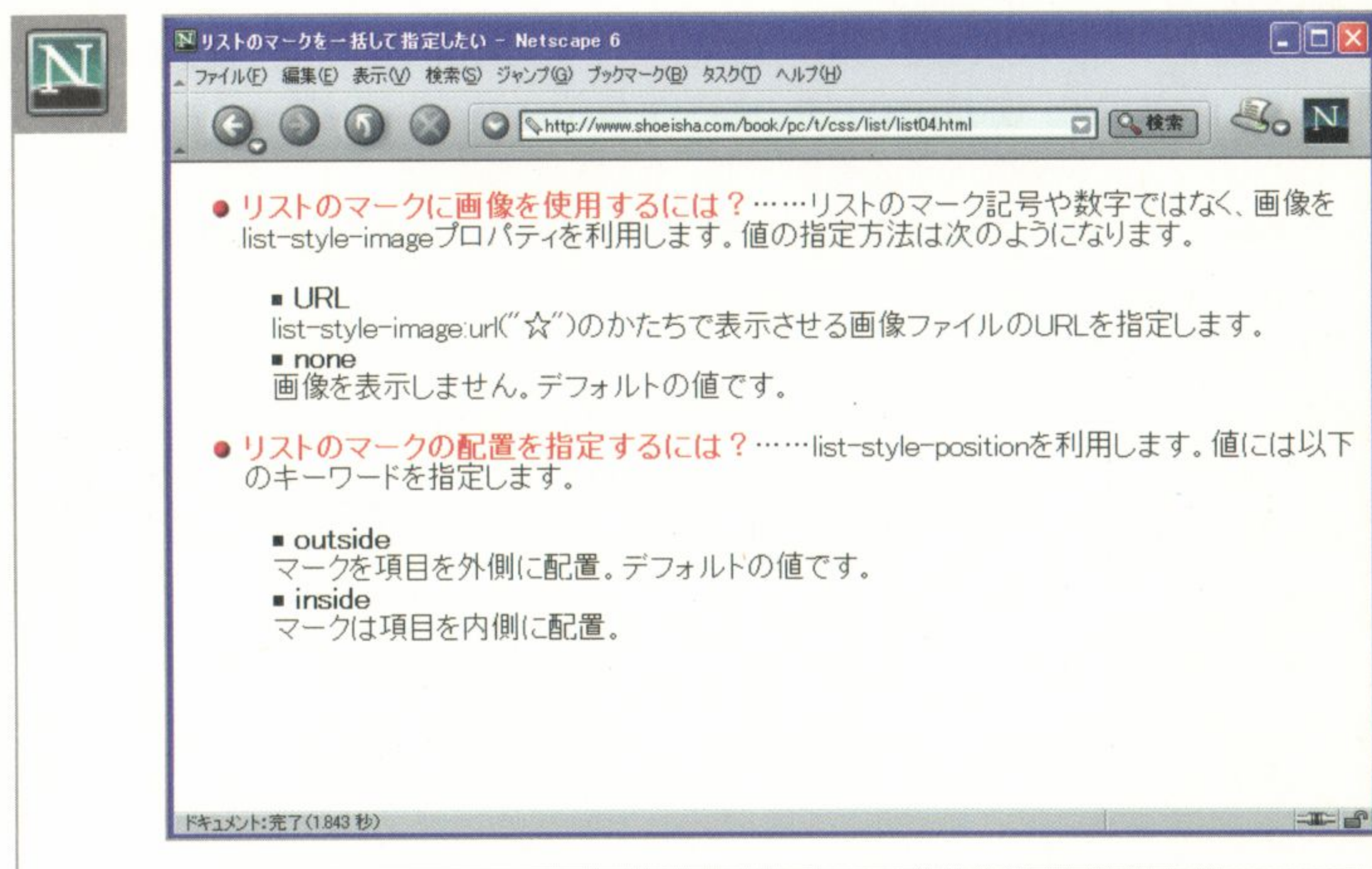
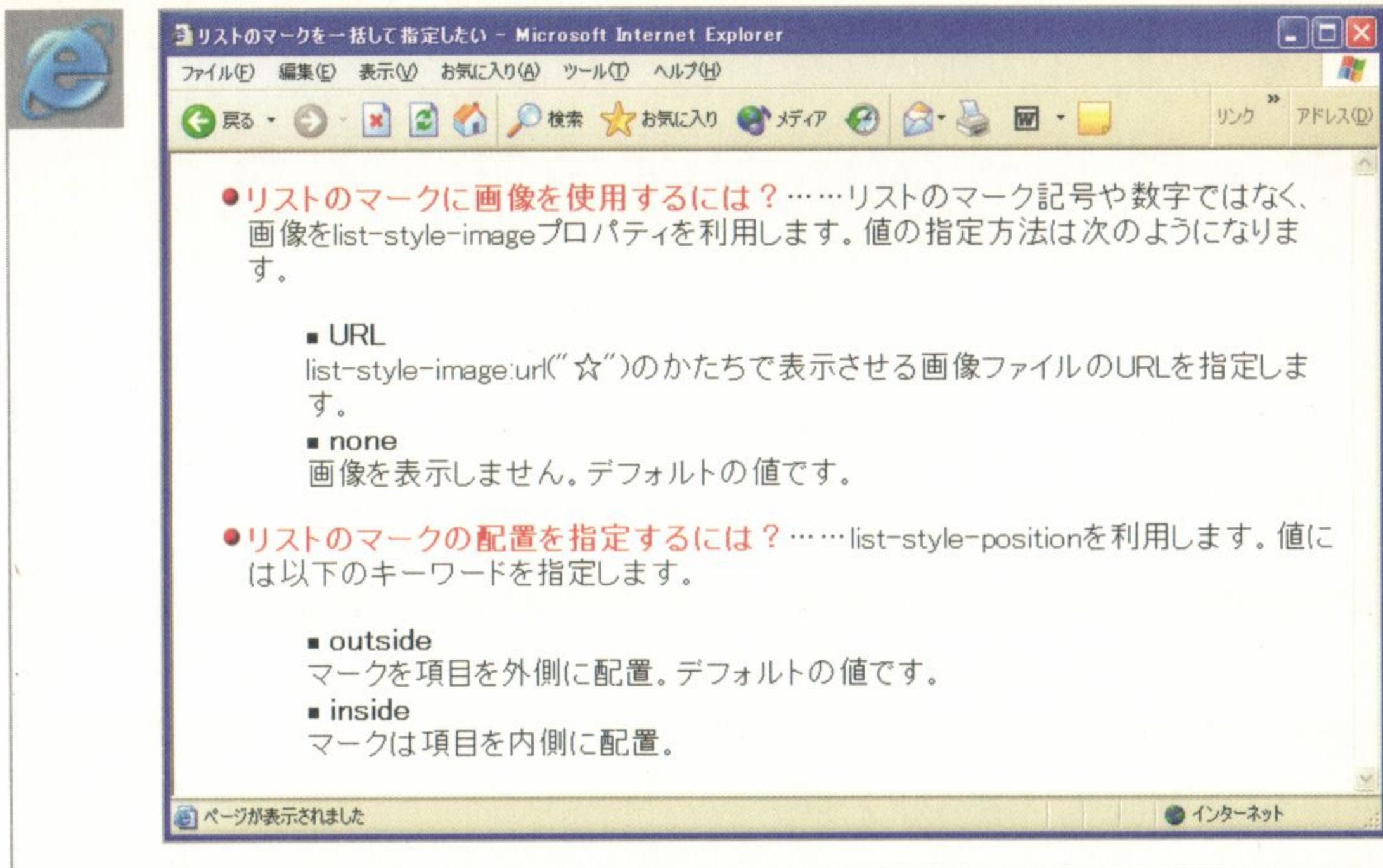
また、list-style プロパティの値に none を指定すると、list-style-type と list-style-image の両方の値が none に設定され、結果としてマーカーが表示されなくなります。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>リストのマークを一括して指定したい</title>
<style type="text/css">
<!--
ul              { margin: 20px auto }
ul.sample1      { list-style: url("ball_red.gif") disc outside }
ul.sample2      {
        list-style-image: none;
        list-style: square inside
 }
.sample3        {
        color: #ff3300;
        font-weight: bold
}
.sample4        { font-weight: bold }
-->
</style>
</head>
<body>
<ul class="sample1">
   <li>
   <span class="sample3">リストのマークに画像を使用するには？</span>……リストのマーク記号や数字ではなく、画像をlist-style-imageプロパティを利用します。値の指定方法は次のようになります。
   </li>
   <ul class="sample2">
      <li>
      <span class="sample4">URL</span><br>list-style-image:url("☆")のかたちで表示させる画像ファイルのURLを指定します。
      </li>
```

```
        <li>
        <span class="sample4">none</span><br>画像を表示しません。デフォルト
        の値です。
        </li>
    </ul>
    <li>
    <span class="sample3">リストのマークの配置を指定するには？</span>……
    list-style-positionを利用します。値には以下のキーワードを指定します。
    </li>
    <ul class="sample2">
        <li>
        <span class="sample4">outside</span><br>マークを項目を外側に配置。デフ
        ォルトの値です。
        </li>
        <li>
        <span class="sample4">inside</span><br>マークは項目を内側に配置。
        </li>
    </ul>
</ul>
</body>
</html>
```

リスト

リスト

IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
○	○	○	○	△*1	△*1	○

＊1：list-style-typeの値のみ有効です

※適用するセレクタによっても効果が変わります

参照
リストのマークを指定したい……………………p.242
リストのマークに画像を使用したい……………p.246
リストのマークの配置を指定したい……………p.248

1 TABLE

テーブルの表示方法を指定したい

table-layout: ★

★……キーワード

テーブルをどのように表示するかを指定します。

通常テーブルはテーブル全体のデータを読み込んでから表示が始まりますが、table-layout プロパティでは最初の横一列分のデータを読み込んだ時点でレイアウトを決定して表示を始めるよう指定することができます。結果として表の表示速度を速めることができます。

値には以下のキーワードがあります。

fixed	テーブルの最初の横一列目のデータを読み込んだ時点でセル幅を計算して表示。width プロパティによってあらかじめセル幅が指定されていない場合（width: auto の場合）には、セル幅は均等になる
auto	テーブル全体を読み込んでからセル幅を計算して表示（デフォルト）

SOURCE

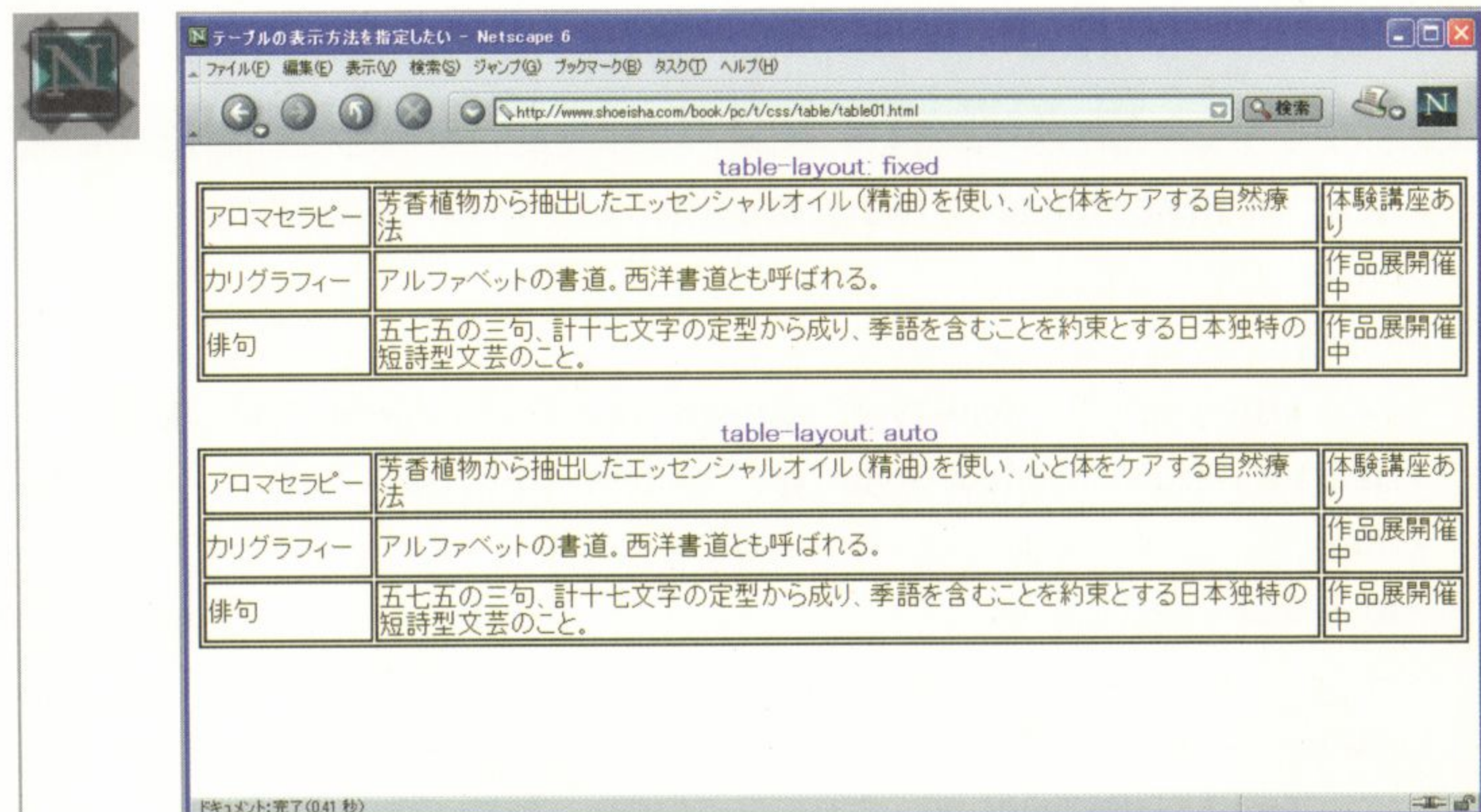
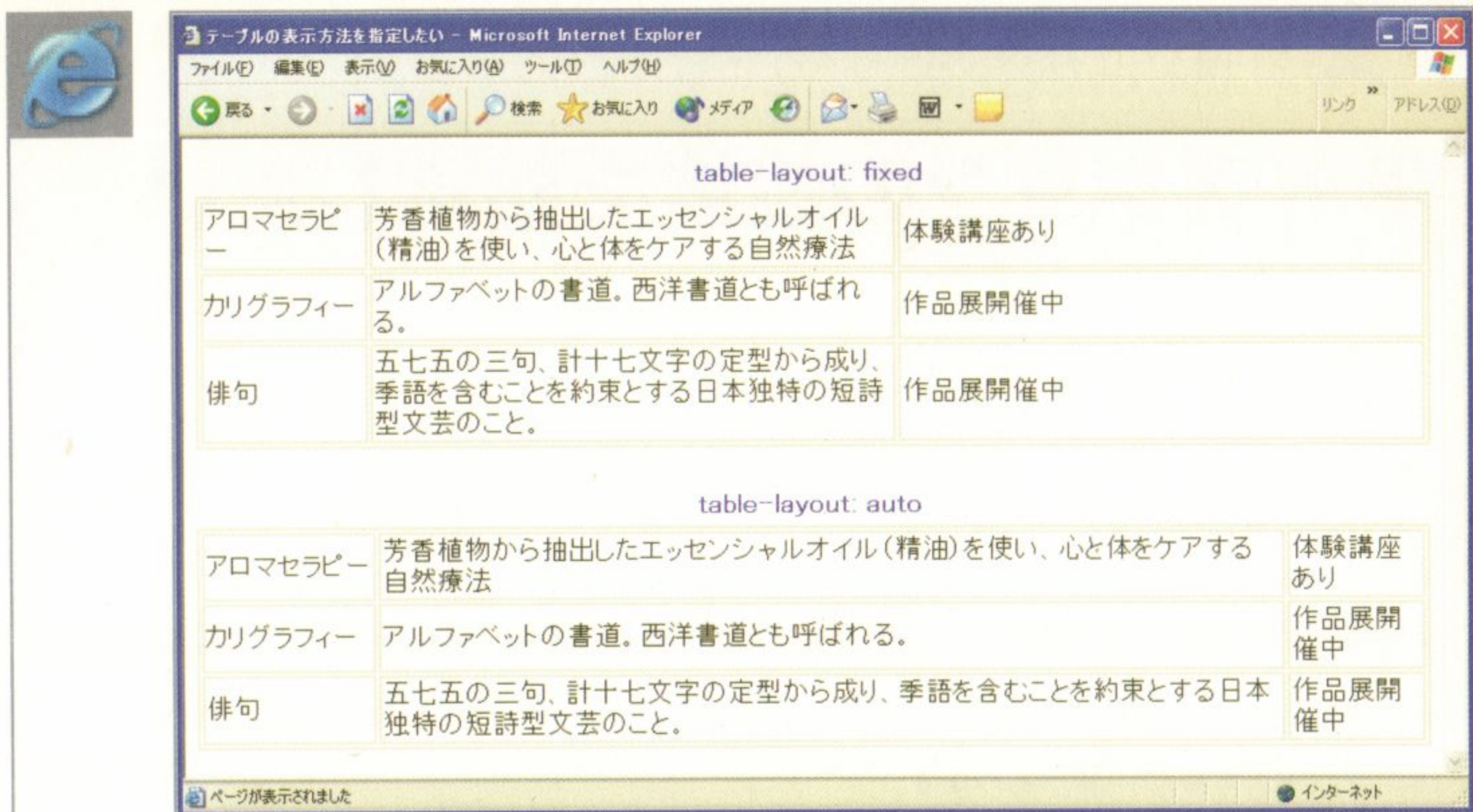
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>テーブルの表示方法を指定したい</title>
<style type="text/css">
<!--
table.sample1 { table-layout: fixed }
table.sample2 { table-layout: auto }
td.sample3     { width: 130px }
caption        {
        font-weight: bold;
        color: #6633ff
}
table,th,td    { border: 2px solid }
div            { margin-bottom: 30px }
-->
</style>
</head>
<body>
<div>
<table class="sample1">
  <caption>table-layout: fixed</caption>
  <tr>
    <td class="sample3">アロマセラピー</td>
    <td>芳香植物から抽出したエッセンシャルオイル（精油）を使い、心と体をケアする自然療法</td>
    <td>体験講座あり</td>
  </tr>
  <tr>
    <td>カリグラフィー</td>
    <td>アルファベットの書道。西洋書道とも呼ばれる。</td>
    <td>作品展開催中</td>
```

テーブル

```
    </tr>
    <tr>
        <td> 俳句 </td>
        <td> 五七五の三句、計十七文字の定型から成り、季語を含むことを約束とする日本
        独特の短詩型文芸のこと。</td>
        <td> 作品展開催中 </td>
    </tr>
</table>
</div>
<div>
<table class="sample2">
    <caption>table-layout: auto</caption>
    <tr>
        <td class="sample3"> アロマセラピー </td>
        <td> 芳香植物から抽出したエッセンシャルオイル（精油）を使い、心と体をケアす
        る自然療法 </td>
        <td> 体験講座あり </td>
    </tr>
    <tr>
        <td> カリグラフィー </td>
        <td> アルファベットの書道。西洋書道とも呼ばれる。</td>
        <td> 作品展開催中 </td>
    </tr>
    <tr>
        <td> 俳句 </td>
        <td> 五七五の三句、計十七文字の定型から成り、季語を含むことを約束とする日本
        独特の短詩型文芸のこと。</td>
        <td> 作品展開催中 </td>
    </tr>
</table>
</div>
</body>
</html>
```

テーブル

▲fixedを指定すると、width指定のないセル幅は均等になります

▲Netscapeは対応していません

テーブル

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
fixed	×	○	○	○	×	×	×
auto	×	○	○	○	×	×	×

※Macintosh版IE5は対応していません

キャプションの位置を指定したい

caption-side: ★

★……キーワード

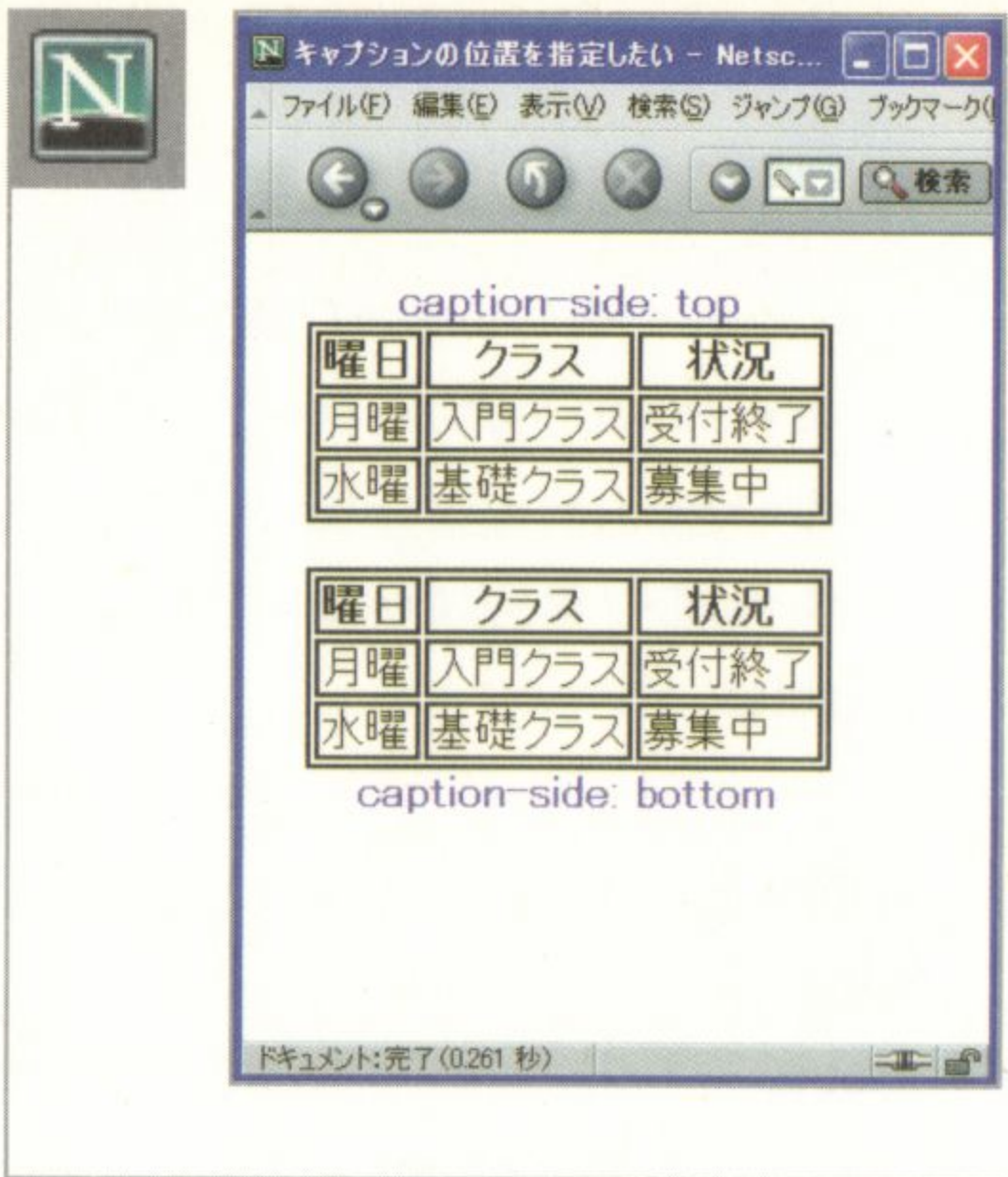
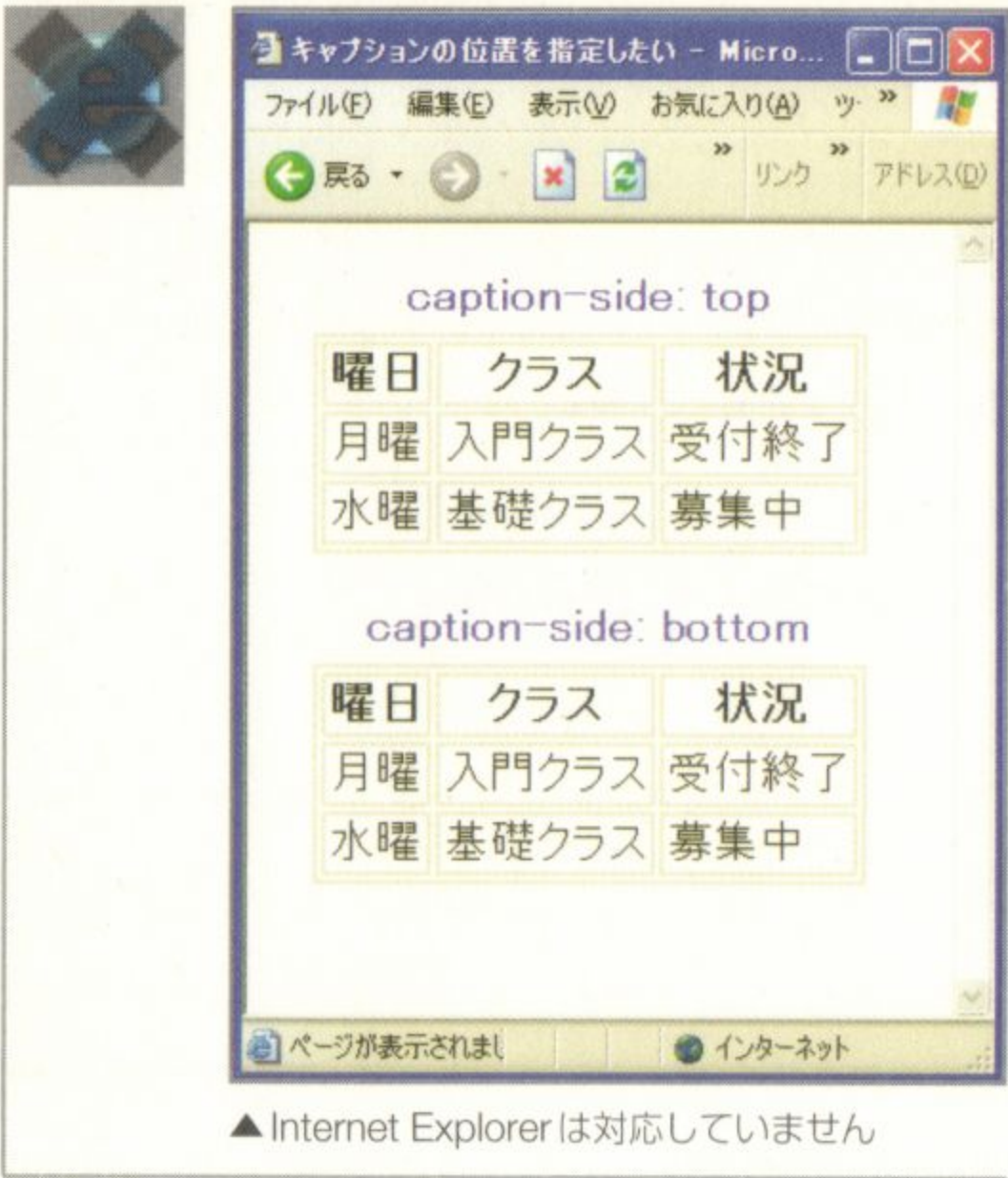
キャプション（タイトル）の表示される位置を指定します。
値には以下のキーワードがあります。

top	テーブルの上部にキャプションを表示（デフォルト）
bottom	テーブルの下部にキャプションを表示

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>キャプションの位置を指定したい</title>
<style type="text/css">
<!--
caption              {
        font-weight: bold;
        color: #6633ff
}
caption#sample1      { caption-side: top }
caption#sample2      { caption-side: bottom }
table,th,td          { border: 2px solid }
div                  { margin:20px }
-->
</style>
</head>
```

```
<body>
<div>
<table>
  <caption id="sample1">caption-side: top</caption>
  <tr><th> 曜日 </th><th> クラス </th><th> 状況 </th></tr>
  <tr><td> 月曜 </td><td> 入門クラス </td><td> 受付終了 </td></tr>
  <tr><td> 水曜 </td><td> 基礎クラス </td><td> 募集中 </td></tr>
</table>

</div>
<div>
<table>
  <caption id="sample2">caption-side: bottom</caption>
  <tr><th> 曜日 </th><th> クラス </th><th> 状況 </th></tr>
  <tr><td> 月曜 </td><td> 入門クラス </td><td> 受付終了 </td></tr>
  <tr><td> 水曜 </td><td> 基礎クラス </td><td> 募集中 </td></tr>
</table>
</div>
</body>
</html>
```

▲Internet Explorerは対応していません

キャプション位置を指定するHTMLタグをCSSに改める

HTMLタグでキャプションの位置を指定するには、次のように<caption>タグにalign属性を指定します。

```
<caption align="top">～</caption>
<caption align="bottom">～</caption>
```

<caption>タグのalign属性はDeprecated（推奨しない）とされており、キャプションの位置はスタイルシートで指定することが推奨されています。同様の効果をスタイルシートで表現すると以下のようになります。

```
caption { caption-side: top }
caption { caption-side: bottom }
```


	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
top	×	×	×	×	×	×	○
bottom	×	×	×	×	×	×	○

行揃えを指定したい ……………………………p.69

3 TABLE

枠線の表示形式を指定したい

border-collapse: ★

★……キーワード

表の外枠やセルの枠線の表示方法を指定します。
値には以下のキーワードがあります。

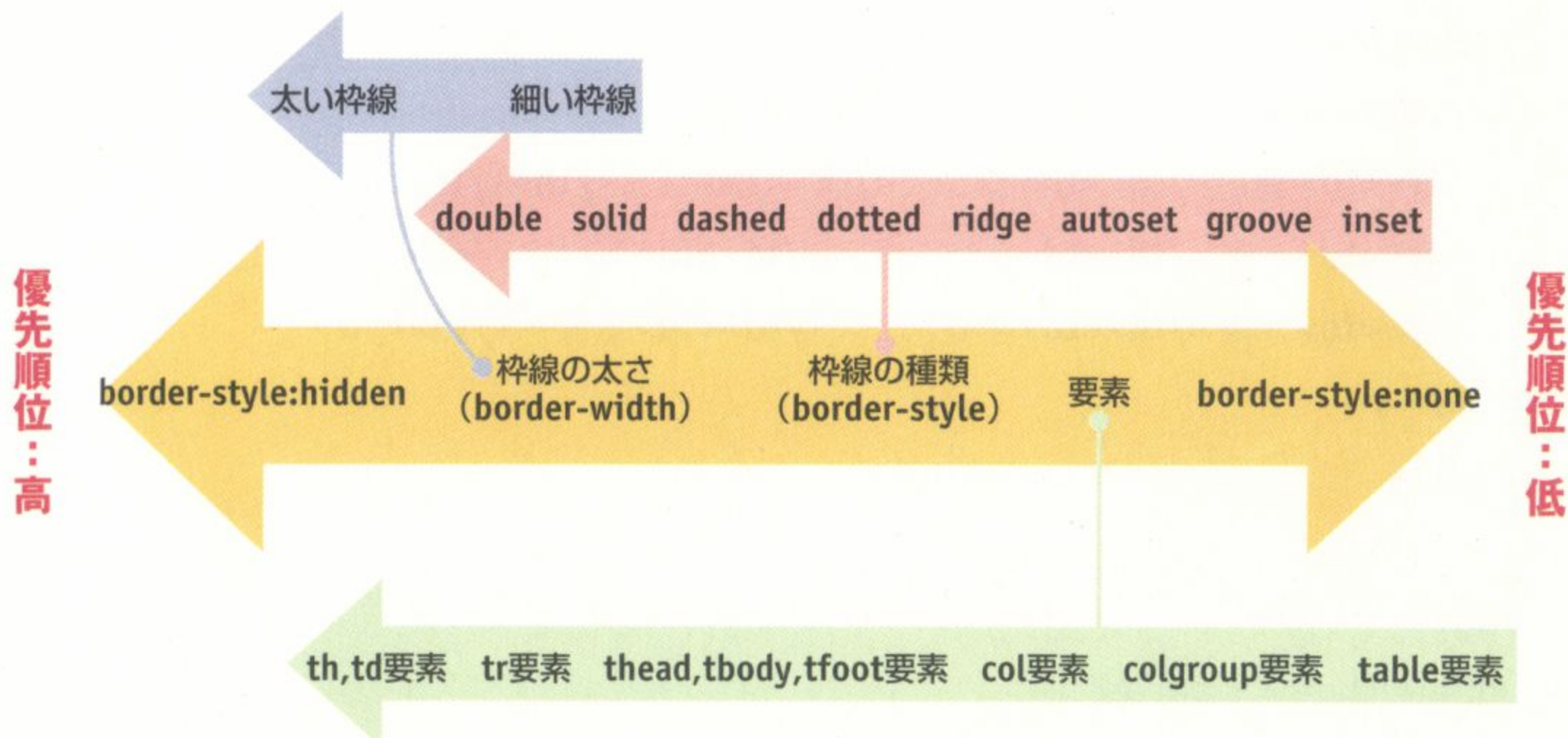
collapse　隣り合うセルの枠線を重ねて表示する
separate　隣り合うセルの枠線を分離させて表示する（デフォルト）

なお、border-collapse プロパティの値に collapse を指定することで形式の異なる枠線が重なった場合の優先順位は次のようになります。

1. border-style: hidden が指定されたものがもっとも優先される
2. border-style: none が指定されたものはもっとも優先度が低い
3. hidden と none 以外の値が指定されている場合は、より幅が太い枠線が優先される
4. 太さも同じ場合は、枠線の種類により以下のような優先度になる
   double ＞ solid ＞ dashed ＞ dotted ＞ ridge ＞ autoset ＞ groove ＞ inset
5. 色（border-color）だけが異なる場合は、以下のような優先度になる
   th 要素・td 要素＞ tr 要素＞ thead 要素・tbody 要素・tfoot 要素＞ col 要素＞ colgroup 要素＞ table 要素

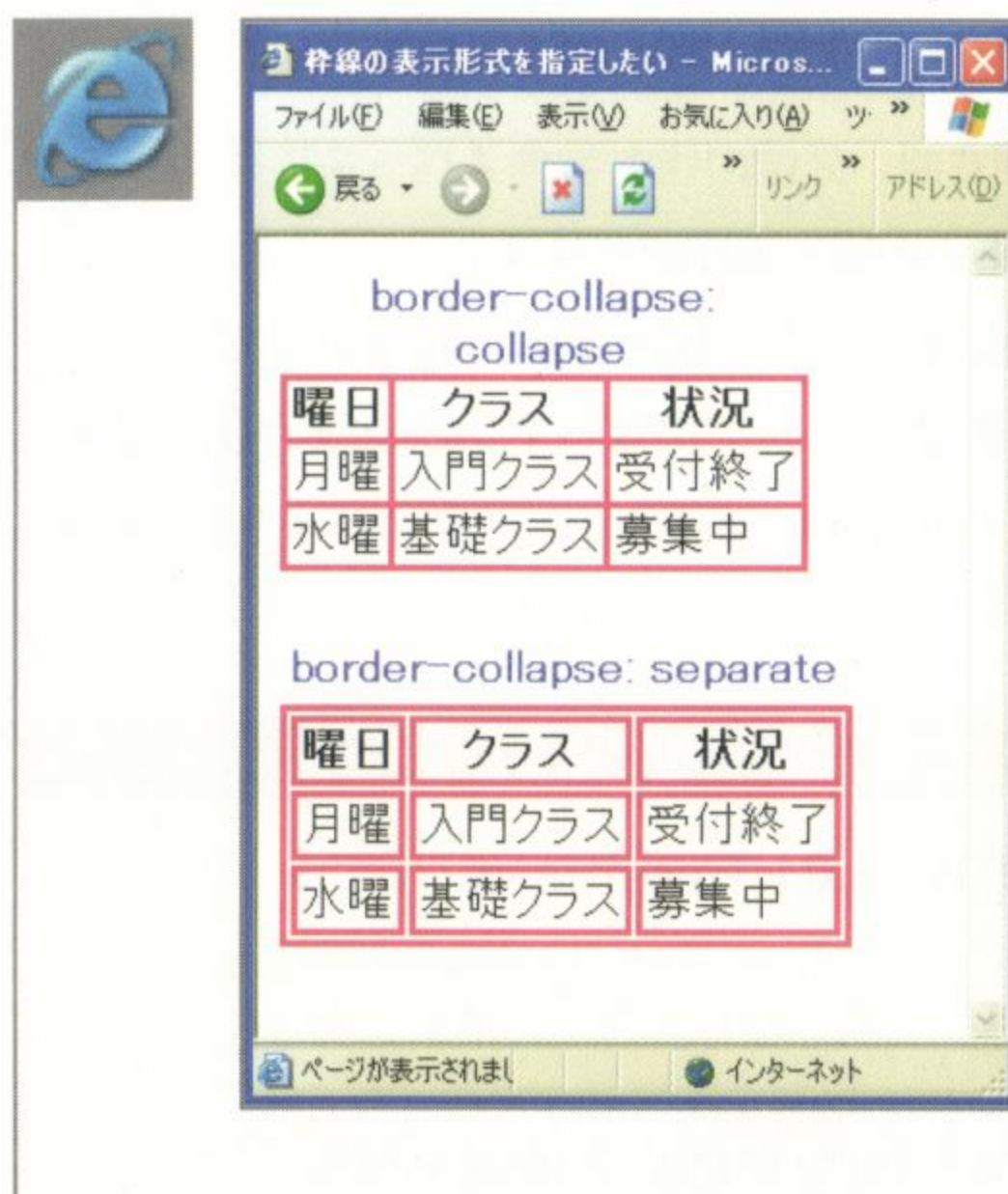
太い枠線　細い枠線
double　solid　dashed　dotted　ridge　autoset　groove　inset
優先順位：高
border-style:hidden　枠線の太さ（border-width）　枠線の種類（border-style）　要素　border-style:none
優先順位：低
th,td要素　tr要素　thead,tbody,tfoot要素　col要素　colgroup要素　table要素

テーブル

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>枠線の表示形式を指定したい</title>
<style type="text/css">
<!--
table.sample1       { border-collapse: collapse }
table.sample2       { border-collapse: separate }
table,th,td         { border: 3px solid #ff0066 }
caption             {
        font-weight: bold;
        color: #6633ff
}
div                 { margin-bottom: 30px }
-->
</style>
</head>

<body>

<div>
<table class="sample1">
   <caption>border-collapse: collapse</caption>
   <tr><th>曜日</th><th>クラス</th><th>状況</th></tr>
   <tr><td>月曜</td><td>入門クラス</td><td>受付終了</td></tr>
   <tr><td>水曜</td><td>基礎クラス</td><td>募集中</td></tr>
</table>
</div>

<div>
<table class="sample2">
   <caption>border-collapse: separate</caption>
```

テーブル

```
    <tr><th>曜日</th><th>クラス</th><th>状況</th></tr>
    <tr><td>月曜</td><td>入門クラス</td><td>受付終了</td></tr>
    <tr><td>水曜</td><td>基礎クラス</td><td>募集中</td></tr>
  </table>
  </div>


  </body>


</html>
```

▲Netscapeは対応していません

テーブル

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
collapse	×	○	○	○	×	×	×
separate	×	○	○	○	×	×	×

※Macintosh版IE5は対応していません

参照
枠線を一括して指定したい……………………p.195
セルの間隔を指定したい……………………p.264

4 TABLE

セルの間隔を指定したい

border-spacing: ★ 上下左右同じ

border-spacing: ★ ★ 左右、上下

★……サイズを表す数値＋単位

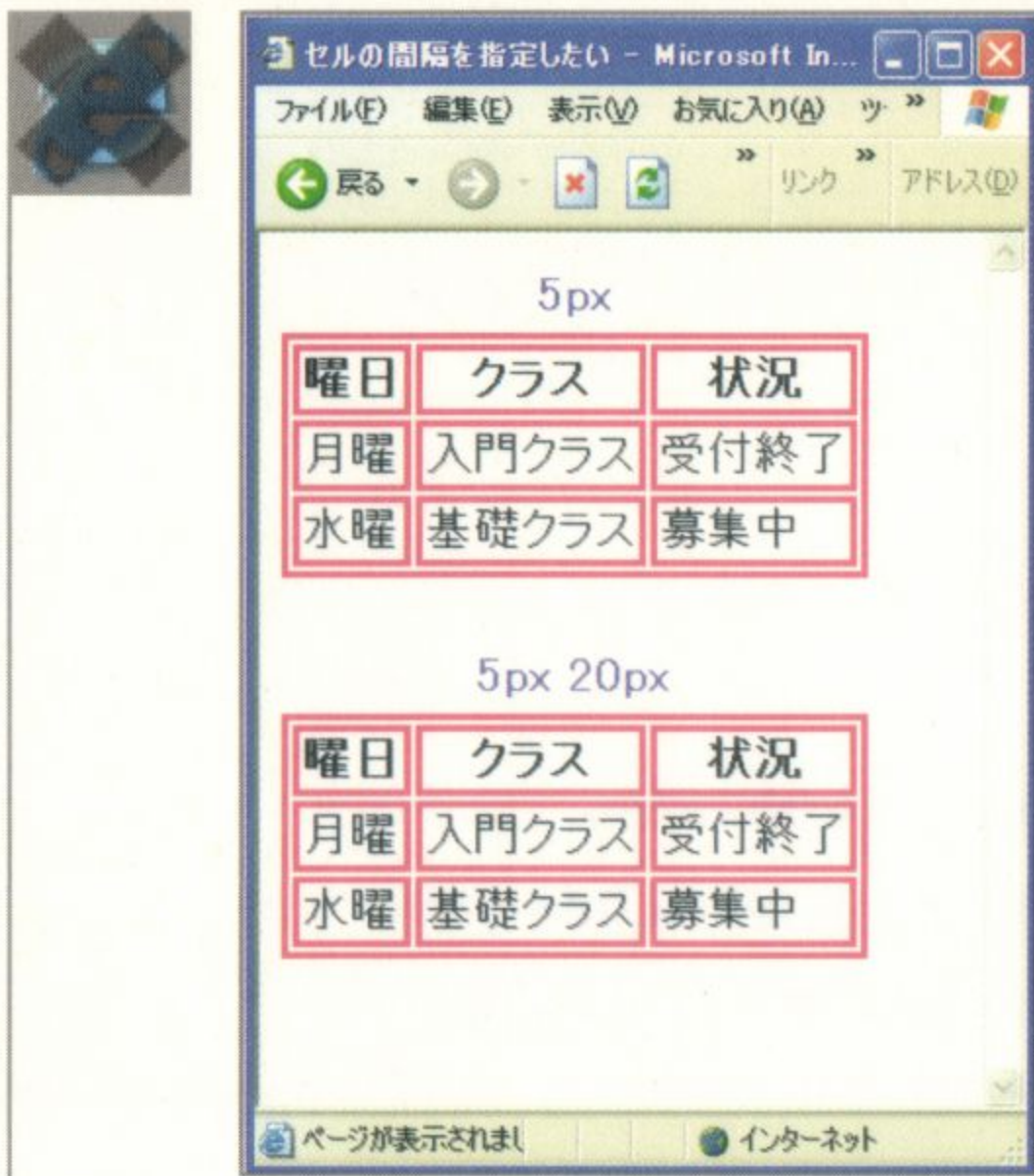
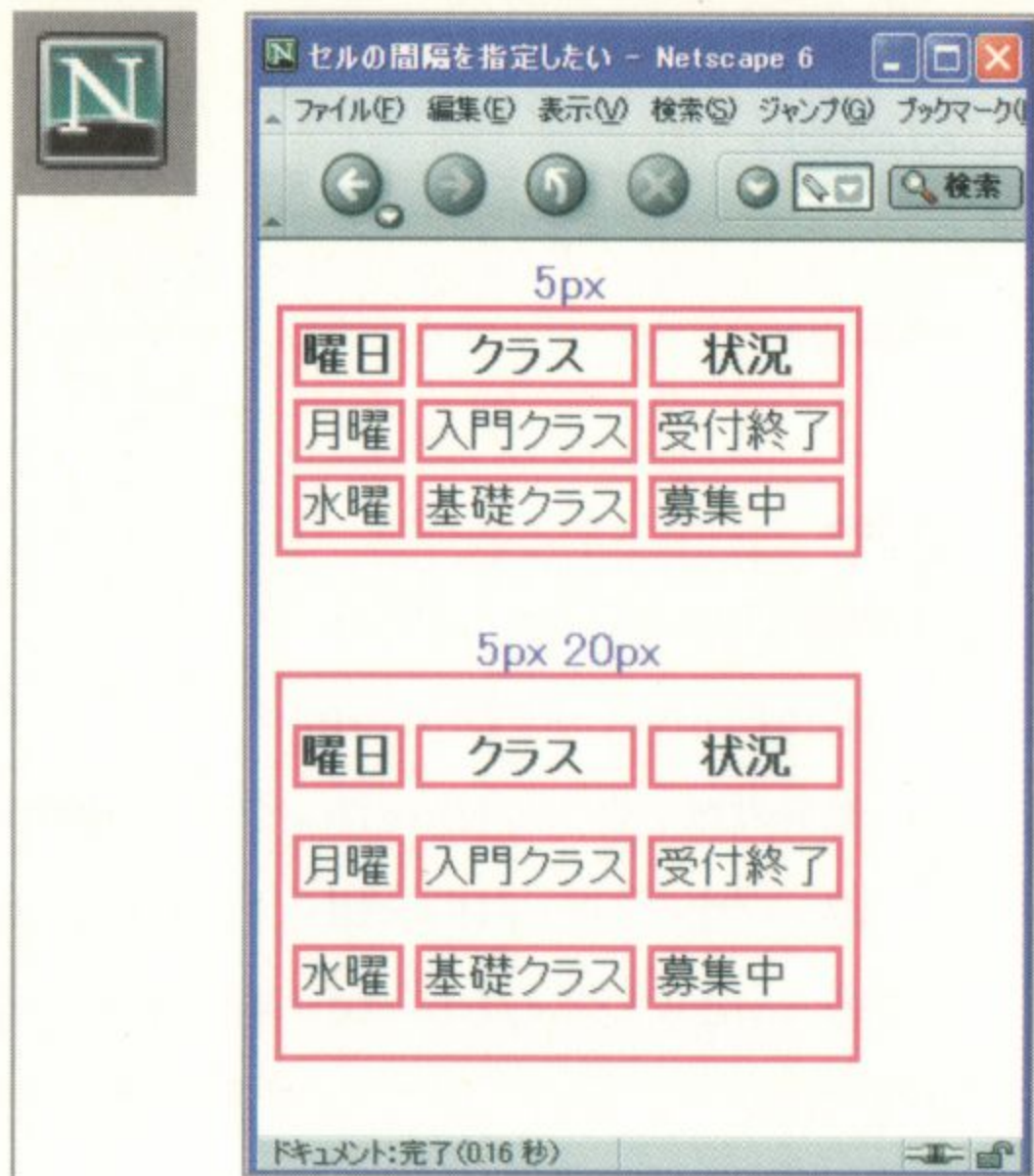
セルの枠線と枠線の間隔を指定します。

値が１つだけの場合は上下左右に同じ間隔が適用され、２個の値を半角スペースで区切って並べた場合には前の値が左右の間隔に、後ろの値が上下の間隔に適用されます。

なお、border-spacingプロパティの指定を有効にするには、border-collapseプロパティの値がseparate（border-collapse:separate/デフォルト）になっている必要があります。border-collapse:collapseが指定されている場合、border-spacingの指定は無効になります。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>セルの間隔を指定したい</title>
<style type="text/css">
<!--
table.sample1     { border-spacing: 5px }
table.sample2     { border-spacing: 5px 20px }
table,th,td       { border: 3px solid #ff0066 }
caption           {
        font-weight: bold;
        color: #6633ff
}
div               { margin-bottom: 30px }
-->
```

```
</style>
</head>

<body>

<div>
<table class="sample1">
   <caption>5px</caption>
   <tr><th>曜日</th><th>クラス</th><th>状況</th></tr>
   <tr><td>月曜</td><td>入門クラス</td><td>受付終了</td></tr>
   <tr><td>水曜</td><td>基礎クラス</td><td>募集中</td></tr>
</table>
</div>

<div>
<table class="sample2">
   <caption>5px 20px</caption>
   <tr><th>曜日</th><th>クラス</th><th>状況</th></tr>
   <tr><td>月曜</td><td>入門クラス</td><td>受付終了</td></tr>
   <tr><td>水曜</td><td>基礎クラス</td><td>募集中</td></tr>
</table>
</div>

</body>

</html>
```

▲Internet Explorerは対応していません

セルの間隔を指定するHTMLタグをCSSに改める

HTMLタグでセルの間隔を調整するには、次のように<table>タグにcellspacing属性を指定します。

<table cellspacing="★">～</table>　　（★ ── 間隔の値）

同様の効果をスタイルシートで表現すると以下のようになります。

table { border-spacing: ★ px }　　（★ ── 間隔の値）

テーブル

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
サイズ	×	×	×	×	×	×	○

参照 枠線の表示形式を指定したい……………………p.261

空セルの枠線の表示方法を指定したい

empty-cells: ★

★……キーワード

空セルの枠線の表示・非表示を指定します。この場合の「空のセル」とは、要素内容が空のセル（データが入っていないセル）だけでなく、visibilityプロパティ（p.206参照）の値にhiddenが指定されているセルも含みます。

値には以下のキーワードがあります。

hide	空セルの枠線を表示しない（デフォルト）
show	空セルの枠線を表示する

なお、empty-cellsプロパティの指定を有効にするには、border-collapseプロパティの値がseparate（border-collapse:separate/デフォルト）になっている必要があります。border-collapse:collapseが指定されている場合、empty-cellsの指定は無効となります。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>空セルの枠線の表示方法を指定したい</title>
<style type="text/css">
<!--
td.sample1     { empty-cells: hide }
td.sample2     { empty-cells: show }
table,th,td    { border: 3px solid #ff0066 }
-->
</style>
</head>
<body>
<table cellspacing="10" cellpadding="10">
        <tr><td>アロマセラピー</td><td class="sample1"></td></tr>
        <tr><td class="sample2"></td><td>カリグラフィー</td></tr>
</table>
</body>
</html>
```

テーブル

▲Internet Explorerは対応していません

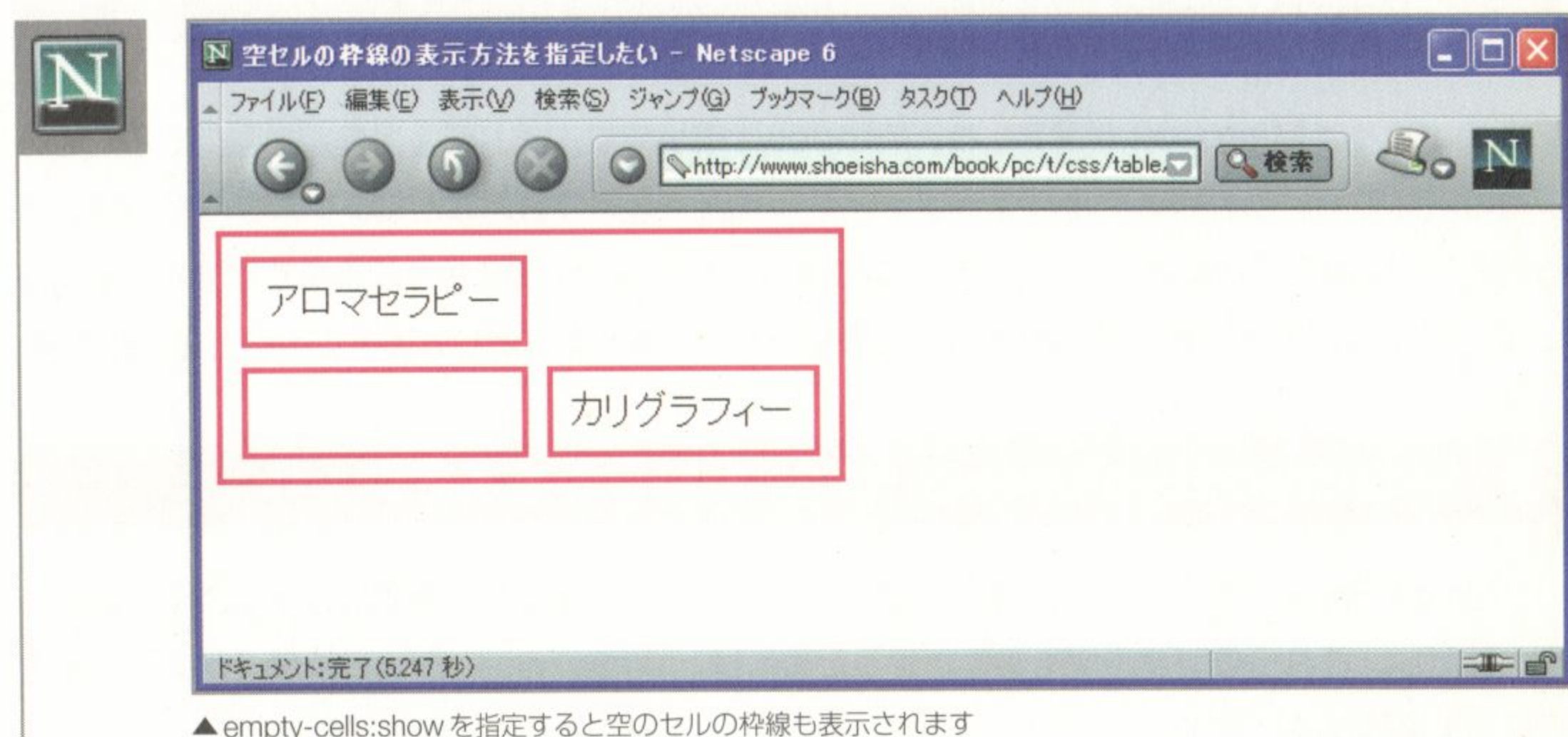
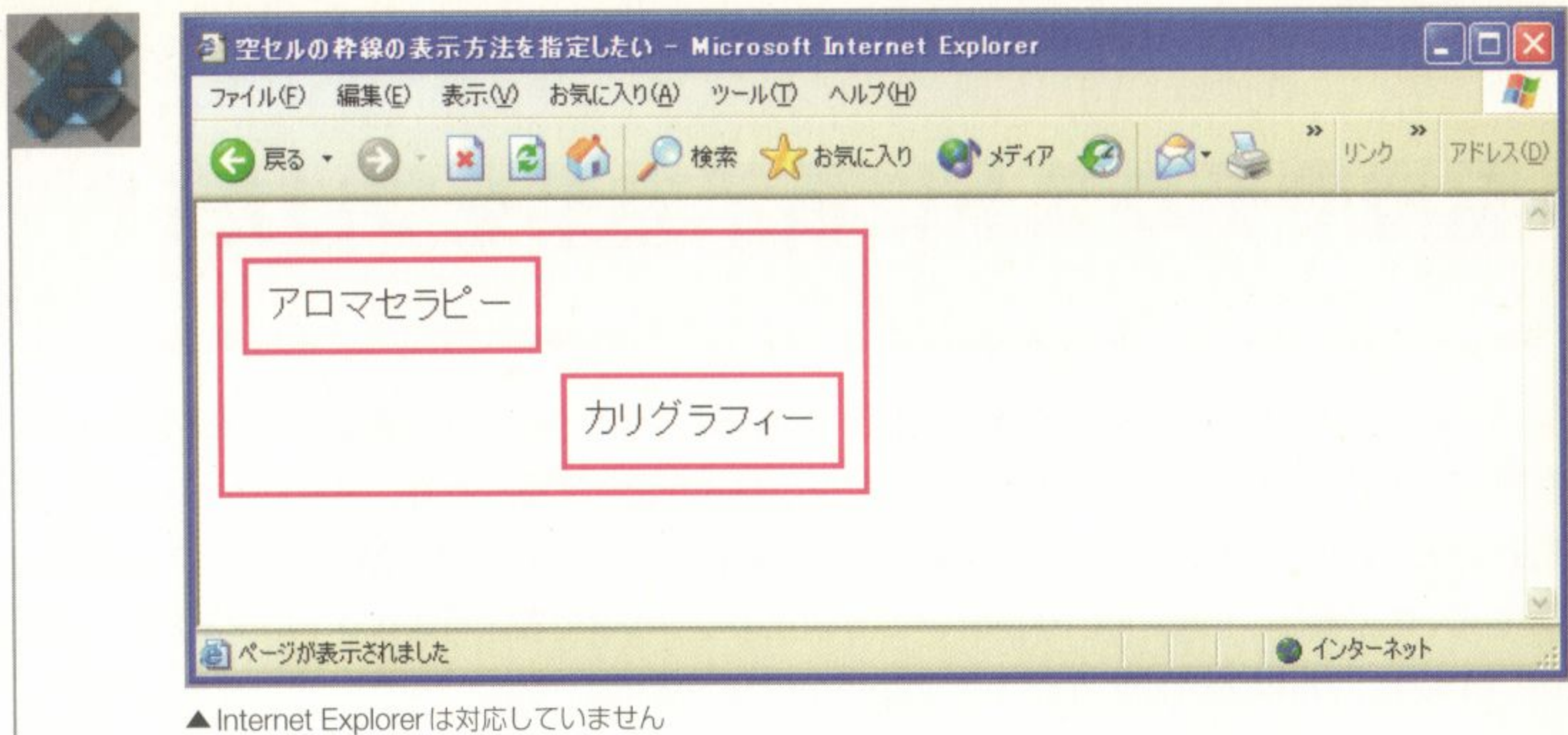
▲empty-cells:showを指定すると空のセルの枠線も表示されます

すべてのセルに枠線を表示するには

table { empty-cells: show }と指定すると、値が継承されてすべてのセルの周囲に枠線が表示されるようになります。

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
hide	×	×	×	×	×	×	○
show	×	×	×	×	×	×	○

※適用するセレクタによっても効果が変わります

参照 枠線の表示形式を指定したい……………………p.261

1 FILTER

画像やテキストに表示効果をつけたい

フィルタとは、Internet Explorerが独自に拡張したWebページ上で視覚的な効果を表現する機能のことで、スタイルシートを利用して定型の書式を記述することで、画像やテキストに特殊効果を設定することができます。Internet Explorer 4.0以上のブラウザで表示可能なフィルタと、DirectXを必要としInternet Explorer 5.5以上のブラウザで表示可能なフィルタとがありますが、いずれもWindows版のInternet Explorerでのみ動作します。

● DirectX

DirectXとはMicrosoftが開発した、Windows環境でマルチメディア機能を強化するための技術の総称です。この技術によりグラフィックス描画や音声の処理などマシンパワーを必要とする処理が、快適で高速に実現できるようになります。Windows 98からは標準で搭載されるようになっていますが、Microsoft社のサイトからダウンロードするなどして入手することも可能です。

● Internet Explorer 5.5以上の場合の基本書式

Internet Explorer 5.5以上の場合、利用できるフィルタの基本的な書式は以下のようになります。ただし、これらのフィルタを動作させるためには、WindowsにDirectXがインストールされている必要があります。

```
filter:progid:DXImageTransform.Microsoft.★(☆)
```

★ ……フィルタ名
☆ ……プロパティ

「filter:progid:DXImageTransform.Microsoft.」につづけて設定したいフィルタ名を、()内には各プロパティとその値を記述します。

● Internet Explorer 4.0 以上の場合の基本書式

Internet Explorer 4.0 以上で利用できるフィルタの基本的な書式は以下のようになります。

filter:★(☆)
★ ……フィルタ名
☆ ……プロパティ

「filter:」につづけてフィルタ名を、()内には各プロパティとその値を記述します。

● フィルタの設定方法

フィルタの設定は、他のスタイルと同様に行います。

次の例では、それぞれ style 属性と style 要素を利用して画像に MotionBlur フィルタを設定しています（MotionBlur フィルタは IE 5.5 以上で利用可能）。MotionBlur フィルタについては p.295 を参照してください。

```
<img src="sample.jpg"
style="filter:progid:DXImageTransform.Microsoft.MotionBlur(strength=50)
width=50%">

<style type="text/css">
img {
    filter:progid:DXImageTransform.Microsoft.MotionBlur(strength=50)
}
</style>
```

フィルタは embed、applet、select、option、tr、thead、tbody、tfoot の各要素をのぞくほぼすべての要素に指定することができます。ただし、これらのフィルタを適用するには要素に対して position プロパティで absolute を指定するか、width プロパティや height プロパティを設定して表示位置を指定するなどの必要がありますので注意してください。また、フィルタによってはボックスを変形させて効果を表現するため、その分の領域も考慮したうえで要素の位置やサイズを指定しておかねばなりません。各フィルタの性質を理解したうえでページのレイアウトを行うことが重要です。

● フィルタの対応

仕様ではさまざまな効果のフィルタが定義されていますが、本書では以下のフィルタを取り上げます。

フィルタ	IE5.5以上	IE4.0以上	備　考	参　照
Alpha	○	○		p.273
Blur	○	×*	＊IE4.0のBlurは性質が異なる	p.276
Chroma	○	○		p.279
DropShadow	○	○		p.282
Glow	○	×		p.285
Emboss	○	×		p.288
Engrave	○	○		p.290
Maskfilter	○	△*	＊IE4.0ではMask	p.292
MotionBlur	○	△*	＊IE4.0ではBlur	p.295
Shadow	○	○		p.298
Wave	○	○		p.301
BasicImage	○	×		p.304
fliph	×*	○	＊BasicImageで表現可能	p.307
flipv	×*	○	＊BasicImageで表現可能	p.307
gray	×*	○	＊BasicImageで表現可能	p.310
invert	×*	○	＊BasicImageで表現可能	p.312
xray	×*	○	＊BasicImageで表現可能	p.314

※すべてWindows版Internet Explorerのみの対応となります

2 FILTER

半透明のフィルタをかけたい

filter:progid:DXImageTransform.Microsoft.Alpha(★) 【IE5.5以上】

filter:alpha(★) 【IE4.0以上】

★……プロパティ

半透明のフィルタをかけます。

指定できるプロパティは次の通りです。

opacity	スタート地点の不透明度（0～100)
finishopacity	終了地点の不透明度（0～100）
style	フィルタの種類（0～3）
startx	スタート地点のＸ座標
starty	スタート地点のＹ座標
finishx	終了地点のＸ座標
finishy	終了地点のＹ座標
enabled	フィルタを実行するかどうか（true、false または 1、0)

opacity と finishopacity で指定する不透明度は、透明の 0 から不透明の 100 までの範囲で整数値で指定します。

style はフィルタの種類を指定します。0 は単一（グラデーションになりません）、1 は扇状、2 は放射線状、3 は長方形に半透明処理を行います。1～3 を指定した場合は、スタート地点と終了地点の座標にしたがってグラデーションが形成されます。

startx と starty はスタート地点、finishx と finishy は終了地点の座標を、それぞれ要素の幅または高さの割合で指定します。デフォルトの値は 0 です。

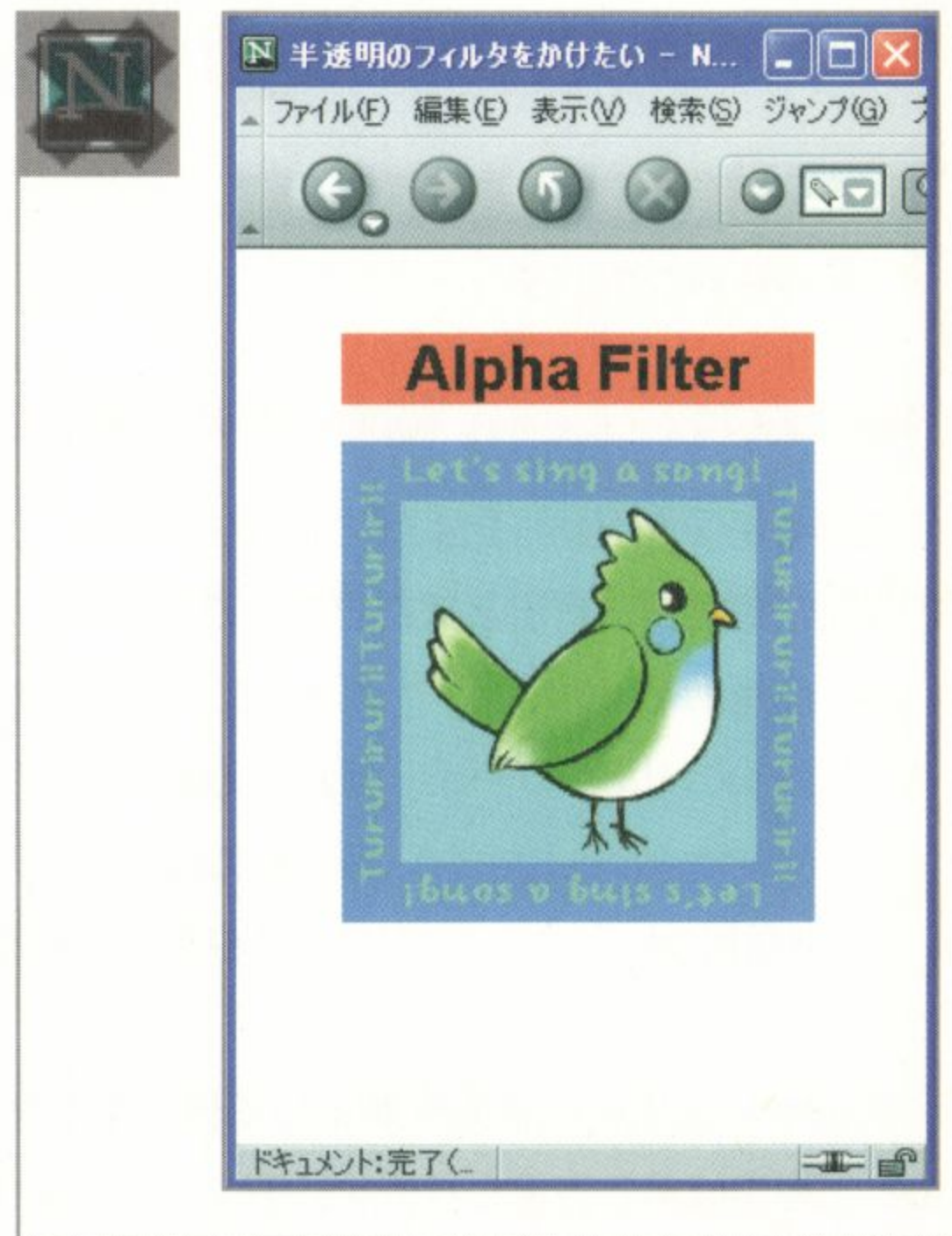
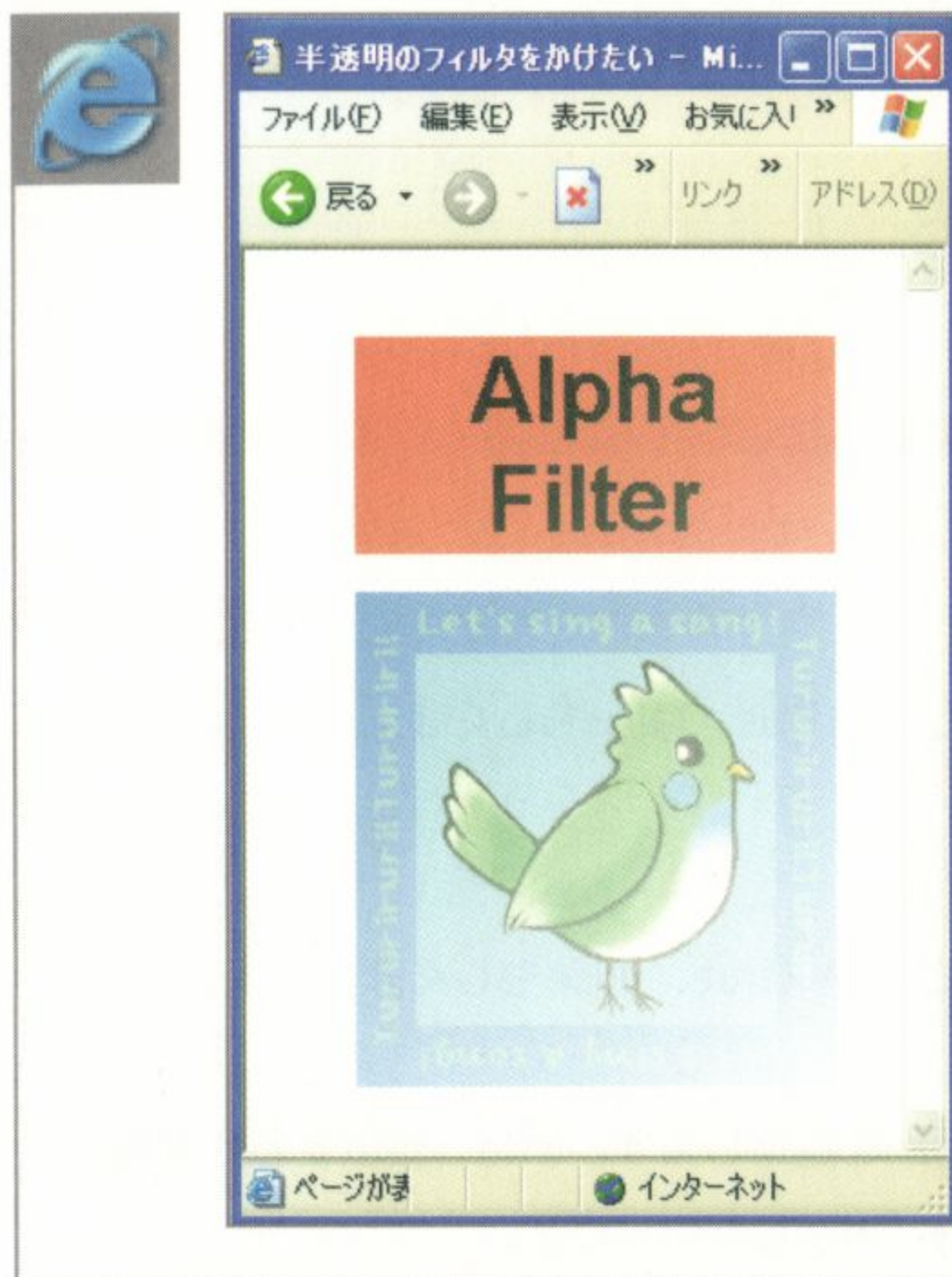
finishopacity、startx、starty、finishx、finishy は style=1 の場合にのみ適用されます。

enabled では、フィルタを実行するかどうかを指定します。true または 1 を指定するとフィルタを実行し、false または 0 を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>半透明のフィルタをかけたい</title>
<style type="text/css">
<!--
.sample        {
        filter:progid:DXImageTransform.Microsoft.Alpha(opacity=100,
        finishopacity=0,style=1,startx=0,starty=0,finishx=100,finishy=100)
}
body           { margin: 0 }
div            {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p              {
        width: 200px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Alpha Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
```

フィルタ

```
</div>
</body>
</html>
```

IE4.0以上のブラウザでの指定方法

IE4.0以上のブラウザで同様のフィルタを表現する場合は次のように指定します。設定方法はAlphaを指定した場合と同じです。

```
.sample {
        filter:alpha(opacity=100,finishopacity=0,style=1,startx=0,starty=0,finishx=100,finishy=100)
}
```

フィルタ

3 FILTER

ぼかしを入れたい

filter:progid:DXImageTransform.Microsoft.Blur(★) 【IE5.5以上】

★……プロパティ

文字や画像をぼかしたように表示します。

指定できるプロパティは次の通りです。

makeshadow	影として表示するかどうか（true、falseもしくは1、0）
pixelradius	ぼかしの量（0～100）
shadowopacity	ぼかしの濃度（0.0～1.0）
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

makeshadowは要素を影として表示するかどうかを指定します。true（表示する）を指定すると、要素は黒で表示されます。デフォルトはfalse（表示しない）です。

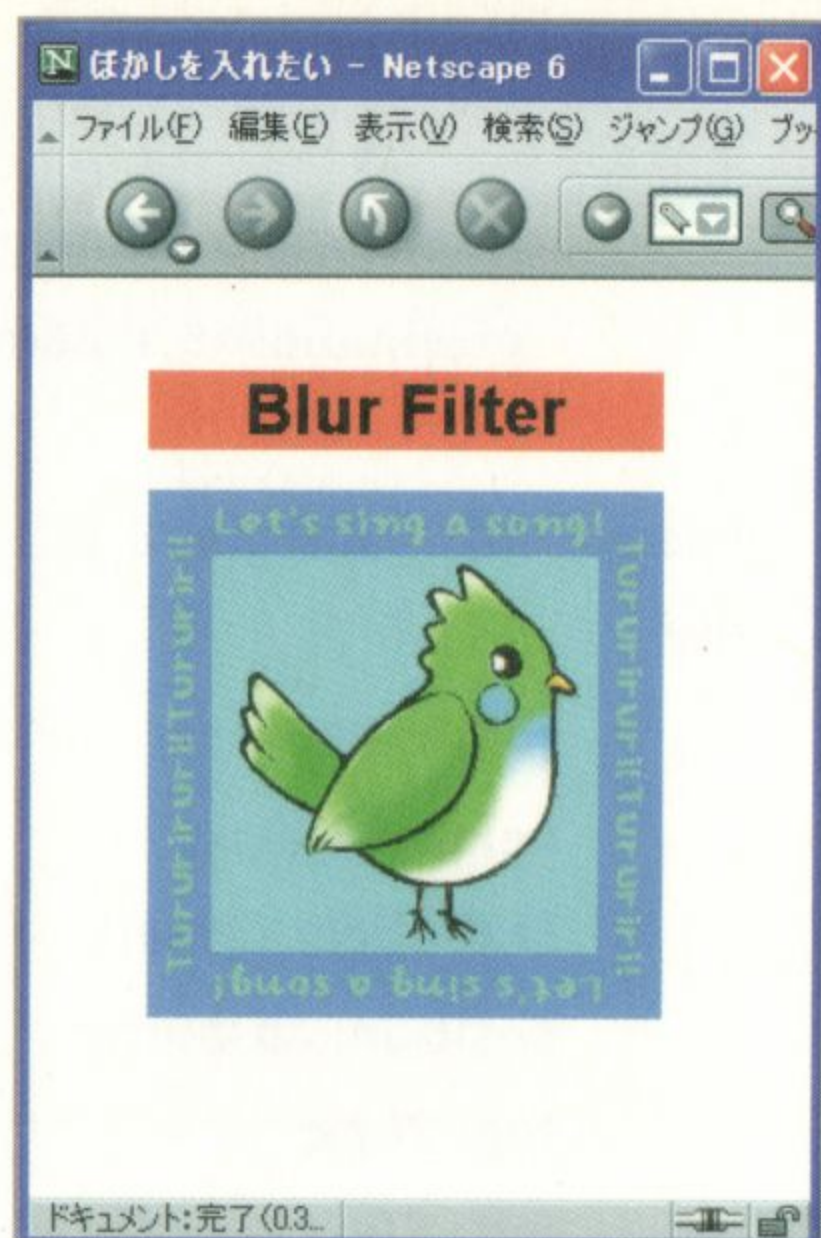
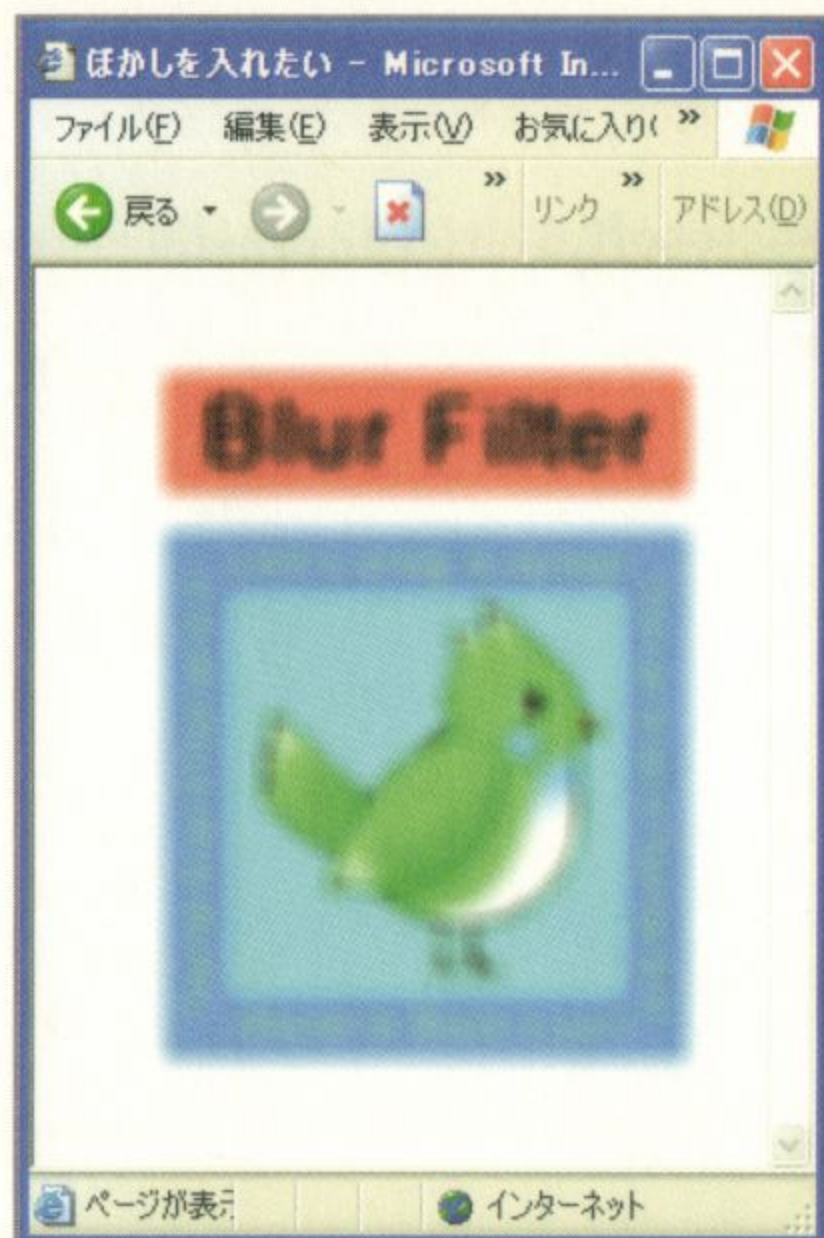
pixelradiusはぼかしの効果が表示される領域を0～100の範囲で指定します。大きな値を指定すると効果を表示するために必要なボックス領域も大きくなるため、サイズ指定やレイアウトに注意が必要です。デフォルトの値は2です。

shadowopacityはぼかしの濃度を0.0～1.0の範囲で指定します。0.75がデフォルトの値です。数値が大きくなるほどぼかしが強くなり、逆に0.0では透明になります。

enabledでは、フィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>ぼかしを入れたい</title>
<style type="text/css">
```

```
<!--
.sample          {
        filter:progid:DXImageTransform.Microsoft.Blur(makeshadow=false,
        pixelradius=5,shadowopacity=0.50)
}
body            { margin: 0 }
div             {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p               {
        width: 200px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Blur Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body>
</html>
```

フィルタ

特色の一色を透過して表示したい

filter:progid:DXImageTransform.Microsoft.Chroma(★) 【IE5.5以上】

filter:chroma(★) 【IE4.0以上】

★……プロパティ

色を透過して表示し、透過GIFのような効果を得ることができます。

指定できるプロパティは次の通りです。

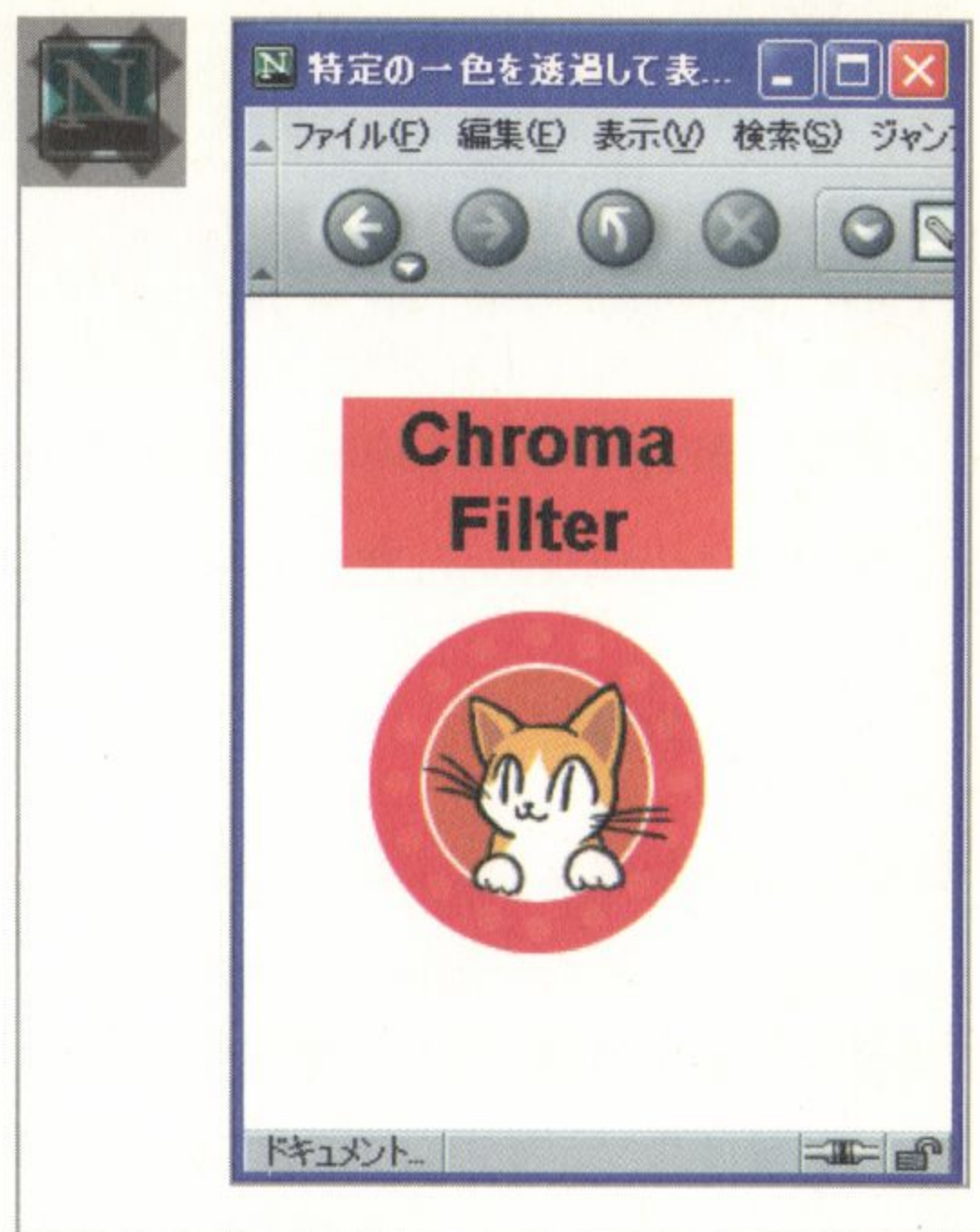
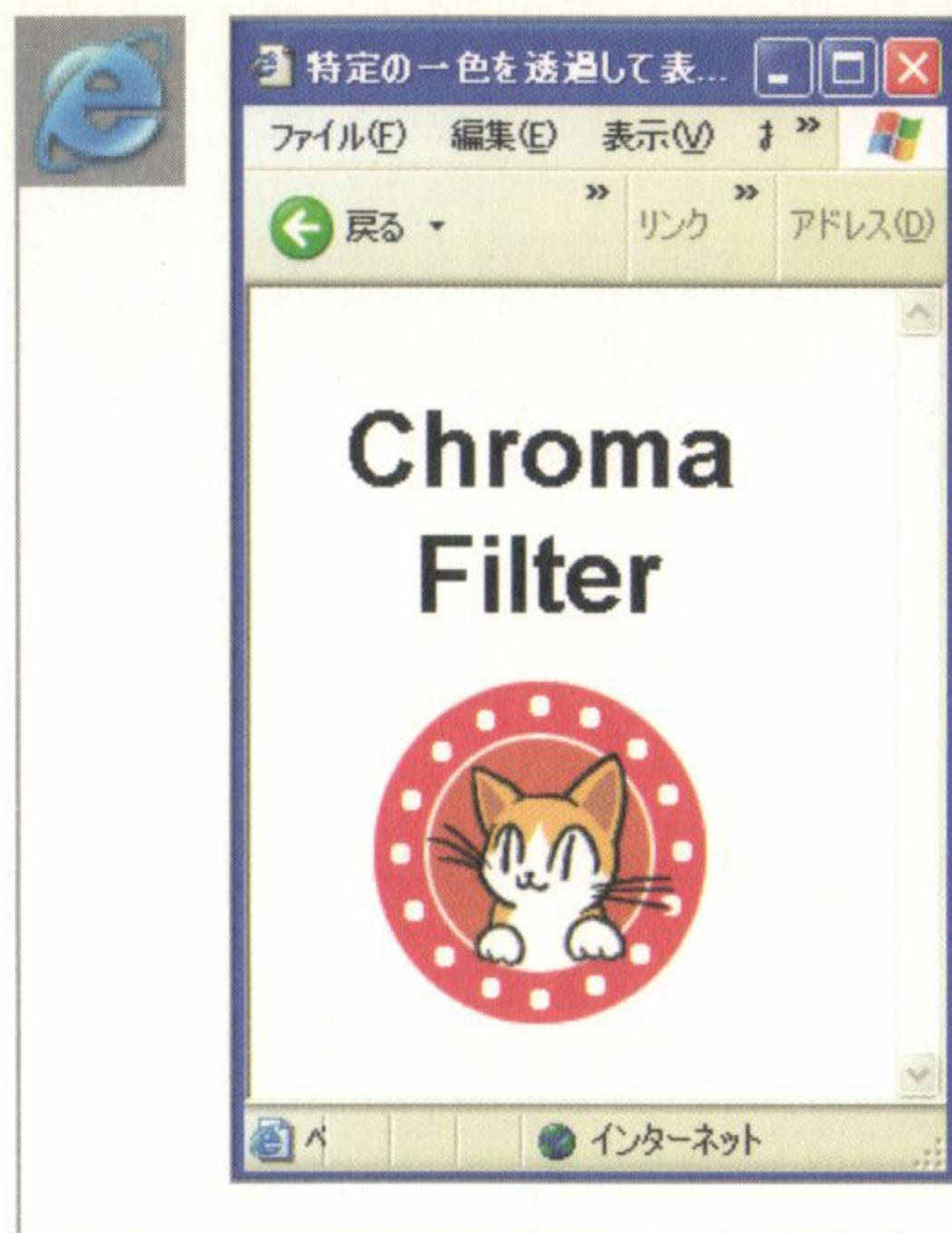
color	透過する色
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

colorで指定した特定の一色を透過して表示します。色の指定には、RGBの数値で指定する方法と、キーワードで指定する方法とがあります。色の詳しい指定方法についてはp.47を参照してください。

enabledでは、フィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>特定の一色を透過して表示したい</title>
<style type="text/css">
<!--
.sample        {
        filter:progid:DXImageTransform.Microsoft.Chroma(color=#de5745)
}
body           { margin: 0 }
div            {
```

```
		width: 170px;
		height: 220px;
		text-align: center;
		position: absolute;
		top: 20px;
		left: 20px
}
p		{
		width: 140px;
		margin: 15px auto;
		font: bold 1.3em Arial,Helvetica,sans-serif;
		background-color: #de5745
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Chroma Filter</p>
<img src="cat.gif" alt="" width="120" height="120">
</div>
</body>
</html>
```

IE4.0以上のブラウザでの指定方法

IE4.0以上のブラウザで同様のフィルタを表現する場合は次のように指定します。設定方法はChromaを指定した場合と同じです。

```
.sample    { filter:Chroma(color=#de5745) }
```

フィルタ

5 FILTER

影をおとしたい

filter:progid:DXImageTransform.Microsoft.DropShadow(★) 【IE5.5以上】

filter:dropshadow(★) 【IE4.0以上】

★••••••プロパティ

影をおとしたような立体的な効果を表現します。

指定できるプロパティは次の通りです。

offx	影の右方向への距離
offy	影の下方向への距離
color	影の色
positive	影の透過の有無（true、falseもしくは1、0）
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

影をおとしたような立体的な効果を表現します。

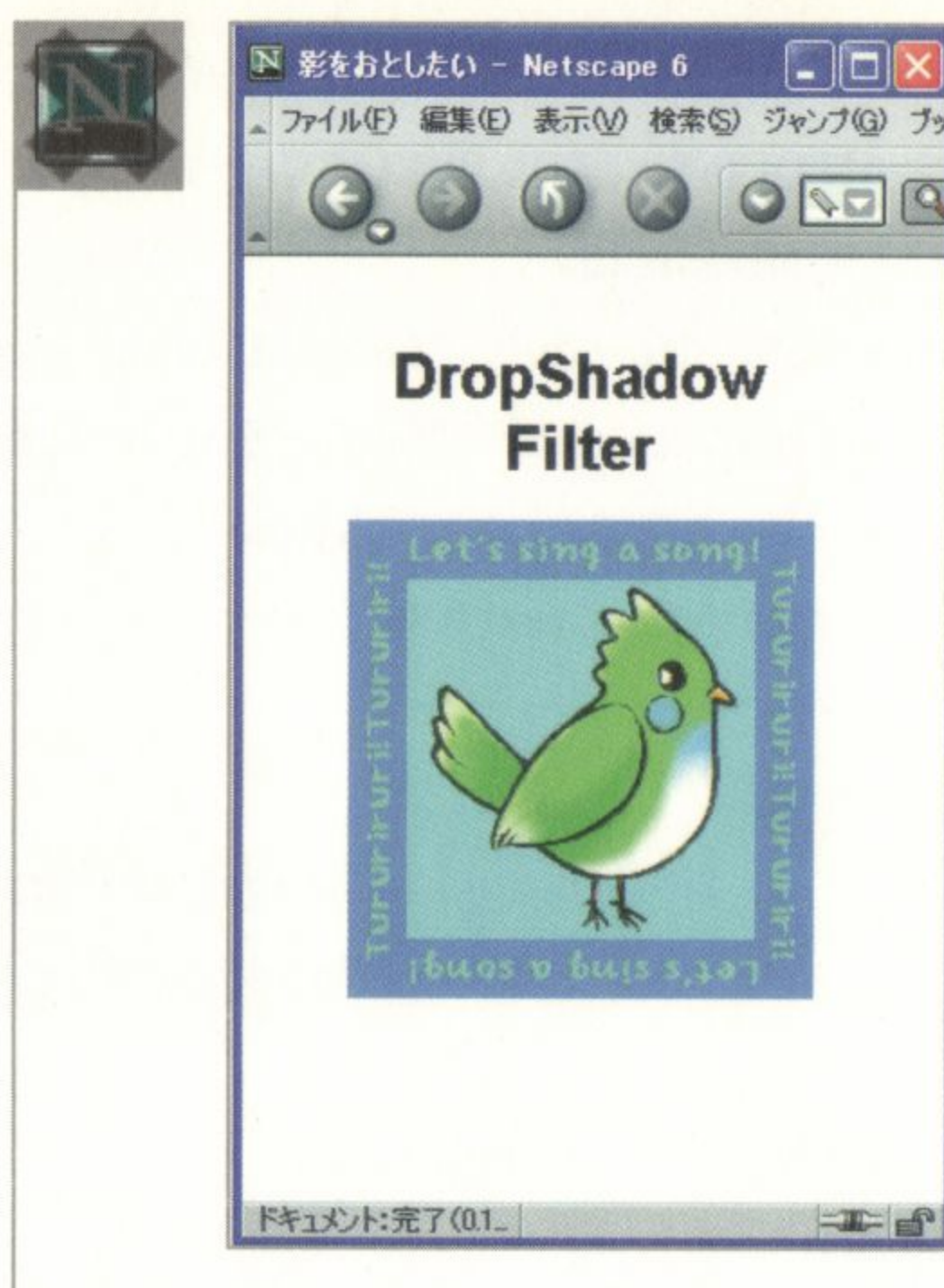
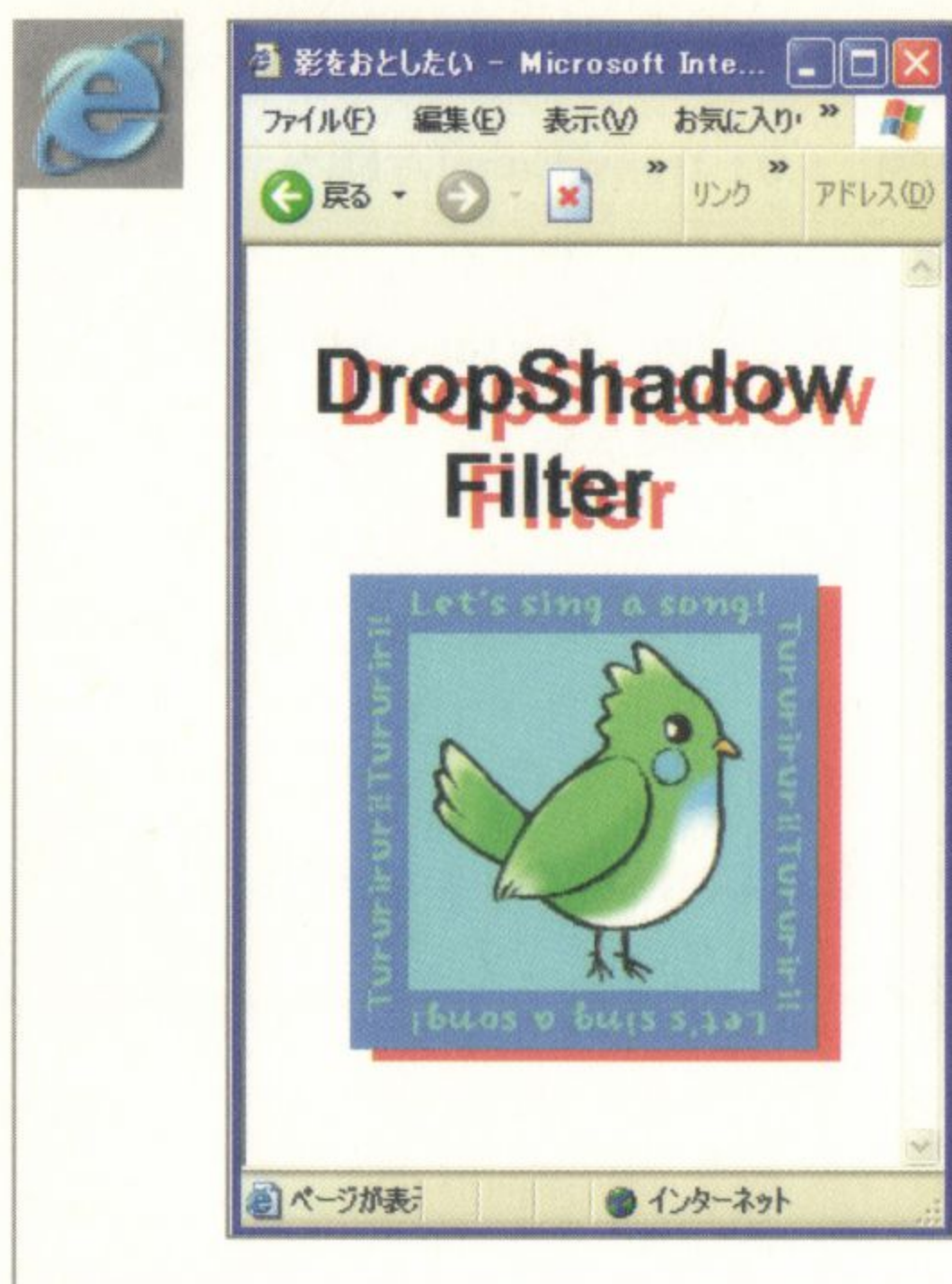
offxで影の右方向への距離を、offyで影の下方向への距離をピクセルで設定します。どちらもマイナスの数値を指定すると左方向、上方向への陰になります。

影の色はcolorで指定します。色の指定には、RGBの数値で指定する方法と、キーワードで指定する方法とがあります。色の詳しい指定方法についてはp.47を参照してください。

positiveは影の透過の有無を設定します。trueまたは1を指定すると不透明な部分を影とし、falseまたは0を指定すると透明な部分を影にします。

enabledは、フィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>影をおとしたい</title>
<style type="text/css">
<!--
.sample         {
        filter:progid:DXImageTransform.Microsoft.DropShadow(offx=10,
        offy=5,color=#ff6633,positive=true)
}
body            { margin: 0 }
div             {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p               {
        width: 200px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif
}
-->
</style>
</head>
<body>
<div class="sample">
<p>DropShadow Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body></html>
```

IE4.0以上のブラウザでの指定方法

IE4.0以上のブラウザで同様のフィルタを表現する場合は次のように指定します。設定方法はDropShadowを指定した場合と同じです。

```
.sample1 {
        filter:dropshadow(offx=10,offy=5,color=#ff6633,positive=true)
}
```

フィルタ

6 FILTER

発光しているように見せたい

filter:progid:DXImageTransform.Microsoft.Glow(★) 【IE5.5以上】

filter:glow(★) 【IE4.0以上】

★……プロパティ

文字や画像の縁から外向きに発光しているような効果を表現します。

指定できるプロパティは次の通りです。

color	発光の色
strength	発光の強さ（1～255）
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

colorで発光色を指定します。色の指定には、RGBの数値で指定する方法と、キーワードで指定する方法とがあります。色の詳しい指定方法についてはp.47を参照してください。

strengthは発光の強さを1～255の整数値で指定します。数値が大きいほど強くなり、発光部分が大きくなります。

enabledはフィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>発光しているように見せたい</title>
<style type="text/css">
<!--
.sample         {
        filter:progid:DXImageTransform.Microsoft.Glow(color=#ff6633,
        strength=10)
}
body            { margin: 0 }
div             {
        width: 170px;
        height: 220px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p               {
        width: 140px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Glow Filter</p>
<img src="cat.gif" alt="" width="120" height="120">
</div>
</body></html>
```

フィルタ

IE4.0以上のブラウザでの指定方法

IE4.0以上のブラウザで同様のフィルタを表現する場合は次のように指定します。設定方法はGlowを指定した場合と同じです。

```
.sample {
        filter:glow(color=#ff6633,strength=10)
}
```

ブラウザによって異なるフィルタ効果領域

フィルタのなかには、効果を表現するために本来の要素サイズよりも大きな空間を周囲に必要とするものがあります。たとえばDropShadowフィルタ（p.282参照）やShadowフィルタ（p.298参照）、Glowフィルタ（p.285参照）などです。

要素の領域が限定されていてフィルタ効果を表現するのに十分な空間がない場合、Internet Explorer 4.0と5.5以上では、処理方法が異なります。IE4.0では領域の外側にはみ出るフィルタ効果部分は切り落として表示しますが、IE5.5以上ではボックス領域のサイズを拡張して全体を表示するため、効果が大きくなるにつれ表示位置がずれることになります。意図したとおりに表現されるページを作成するためには、フィルタ効果で要求される領域も考慮したうえで要素の位置やサイズを指定したり、ページ全体のレイアウトを考えたりする必要があります。

7 FILTER

浮き出したように表示したい

filter:progid:DXImageTransform.Microsoft.Emboss(★)【IE5.5以上】

★••••••プロパティ

文字や画像をグレースケールでエンボス加工（凹凸で浮き上がったように加工する）をしたように表示します。

指定できるプロパティは次の通りです。

bias	浮き上がる高さ（-1.0～1.0）
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

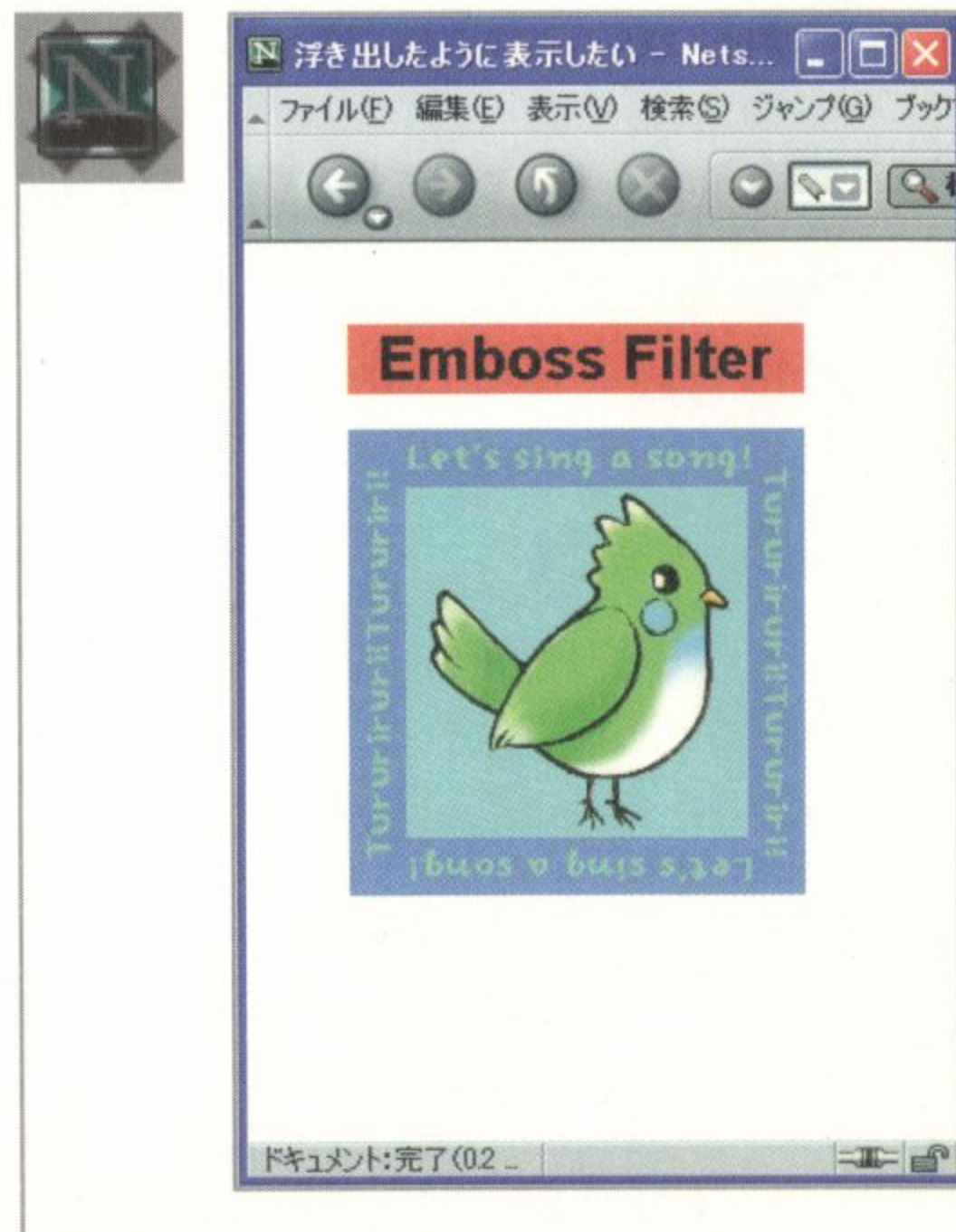
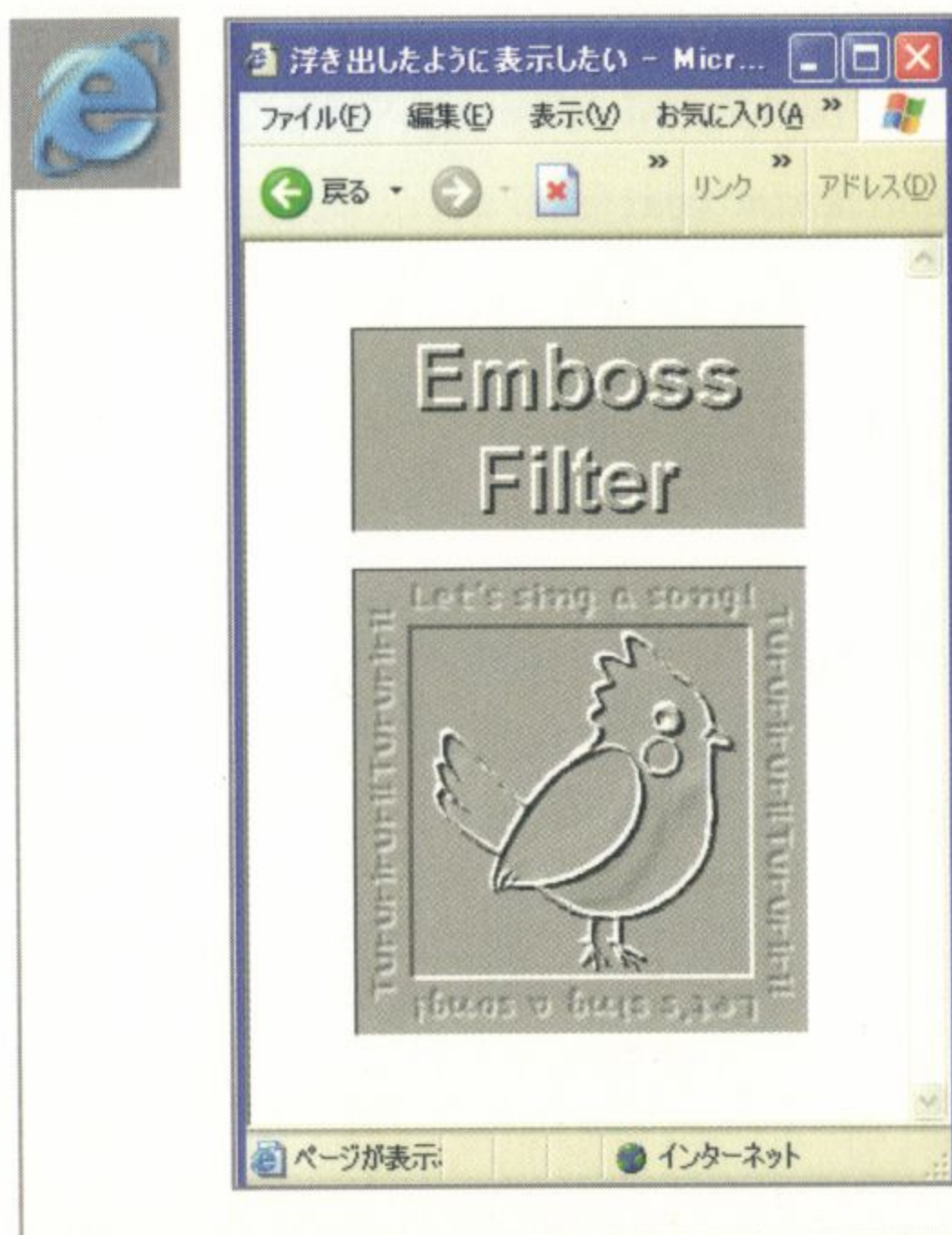
biasは浮き上がる高さを-1.0～1.0の範囲で指定します。デフォルトの値は0.7です。

enabledはフィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>浮き出したように表示したい</title>
<style type="text/css">
<!--
.sample        {
        filter:progid:DXImageTransform.Microsoft.Emboss(bias=0.8)
}
body           { margin: 0 }
div            {
        width: 250px;
        height: 280px;
```

フィルタ

```
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p              {
        width: 200px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Emboss Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body>
</html>
```

フィルタ

8 FILTER

彫り込んだように表示したい

filter:progid:DXImageTransform.Microsoft.Engrave(★)　【IE5.5以上】

★……プロパティ

文字や画像をグレースケールで彫り込んだように表示します。

指定できるプロパティは次の通りです。

bias	へこむ深さ（-1.0～1.0）
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

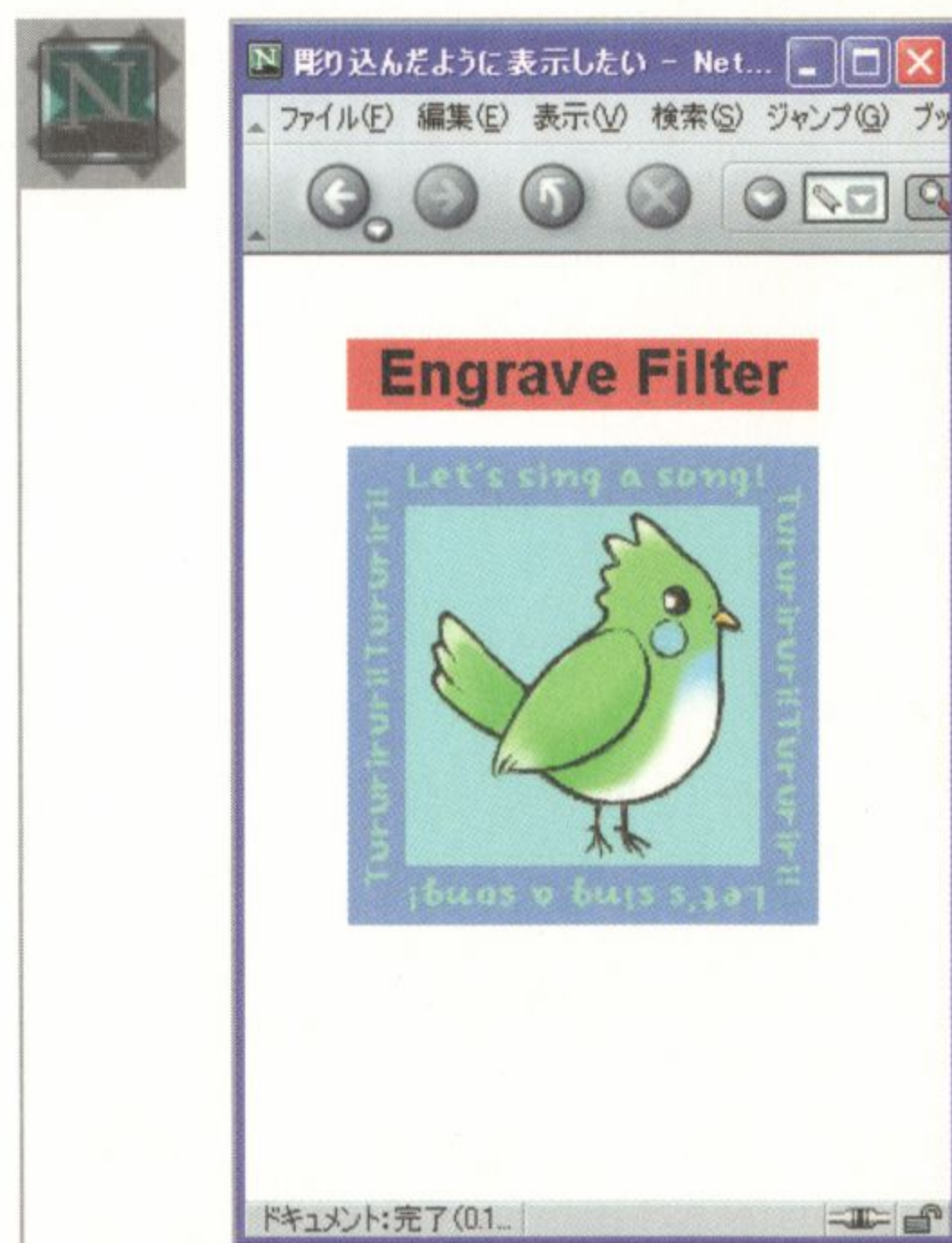
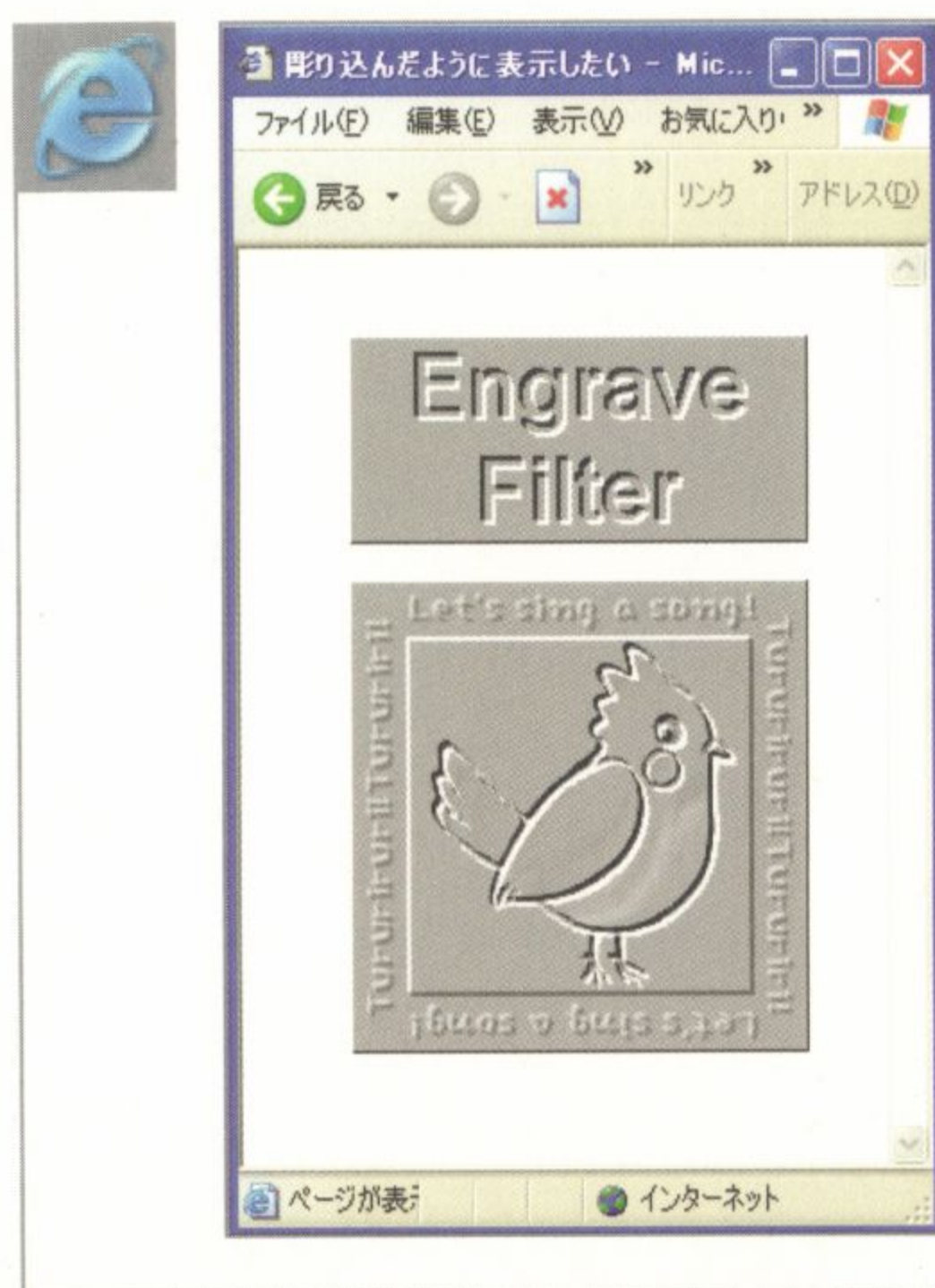
biasはへこむ深さを-1.0～1.0の範囲で指定します。デフォルトの値は0.7です。

enabledはフィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>彫り込んだように表示したい</title>
<style type="text/css">
<!--
.sample       {
        filter:progid:DXImageTransform.Microsoft.Engrave(bias=0.8)
}
body          { margin: 0 }
div           {
        width: 250px;
        height: 280px;
        text-align: center;
```

フィルタ

```
        position: absolute;
        top: 20px;
        left: 20px
}
p               {
        width: 200px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Engrave Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body>
</html>
```

フィルタ

9 FILTER

マスクをかけたい

filter:progid:DXImageTransform.Microsoft.MaskFilter(★)　【IE5.5以上】

filter:mask(★)　【IE4.0以上】

★……プロパティ

文字や画像をマスク処理をして表示します。

指定できるプロパティは次の通りです。

color	塗りつぶす色
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

文字や画像の不透明部分を切り抜いて透明にし、背景や透過処理をした部分など本来透明な部分をcolorプロパティで指定した色で塗りつぶします。色の指定には、RGBの数値で指定する方法と、キーワードで指定する方法とがあります。色の詳しい指定方法についてはp.47を参照してください。

enabledはフィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>マスクをかけたい</title>
<style type="text/css">
<!--
.sample       {
    filter:progid:DXImageTransform.Microsoft.MaskFilter(color=#ff6633)
}
body          { margin: 0 }
div           {
        width: 170px;
        height: 220px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p             {
        width: 140px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif
}
-->
</style>
</head>
<body>
<div class="sample">
<p>MaskFilter Filter</p>
<img src="cat.gif" alt="" width="120" height="120">
</div>
</body>
</html>
```

フィルタ

IE4.0以上のブラウザでの指定方法

IE4以上のブラウザで同様のフィルタを表現する場合は次のように指定します。設定方法はMaskFilterを指定した場合と同じです。

```
.sample { filter:mask(color=#ff6633) }
```

フィルタ

10 FILTER

ブレをつけたい

filter:progid:DXImageTransform.Microsoft.MotionBlur(★) 【IE5.5以上】
filter:blur(★) 【IE4.0以上】

★……プロパティ

文字や画像をブレたように表示します。

指定できるプロパティは次の通りです。

direction	ブレの方向（45度単位）
strength	ブレの強さ
add	元の画像を合成するかどうか（true、falseもしくは1、0）
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

directionはブレの方向を45度単位で指定します。たとえば、0で真上へ、180で真下の方向へのブレとなります。デフォルトは270で左の方向となります。

directionの値	0	45	90	135	180	225	270	315
ブレの方向	上	右上	右	右下	下	左下	左	左上

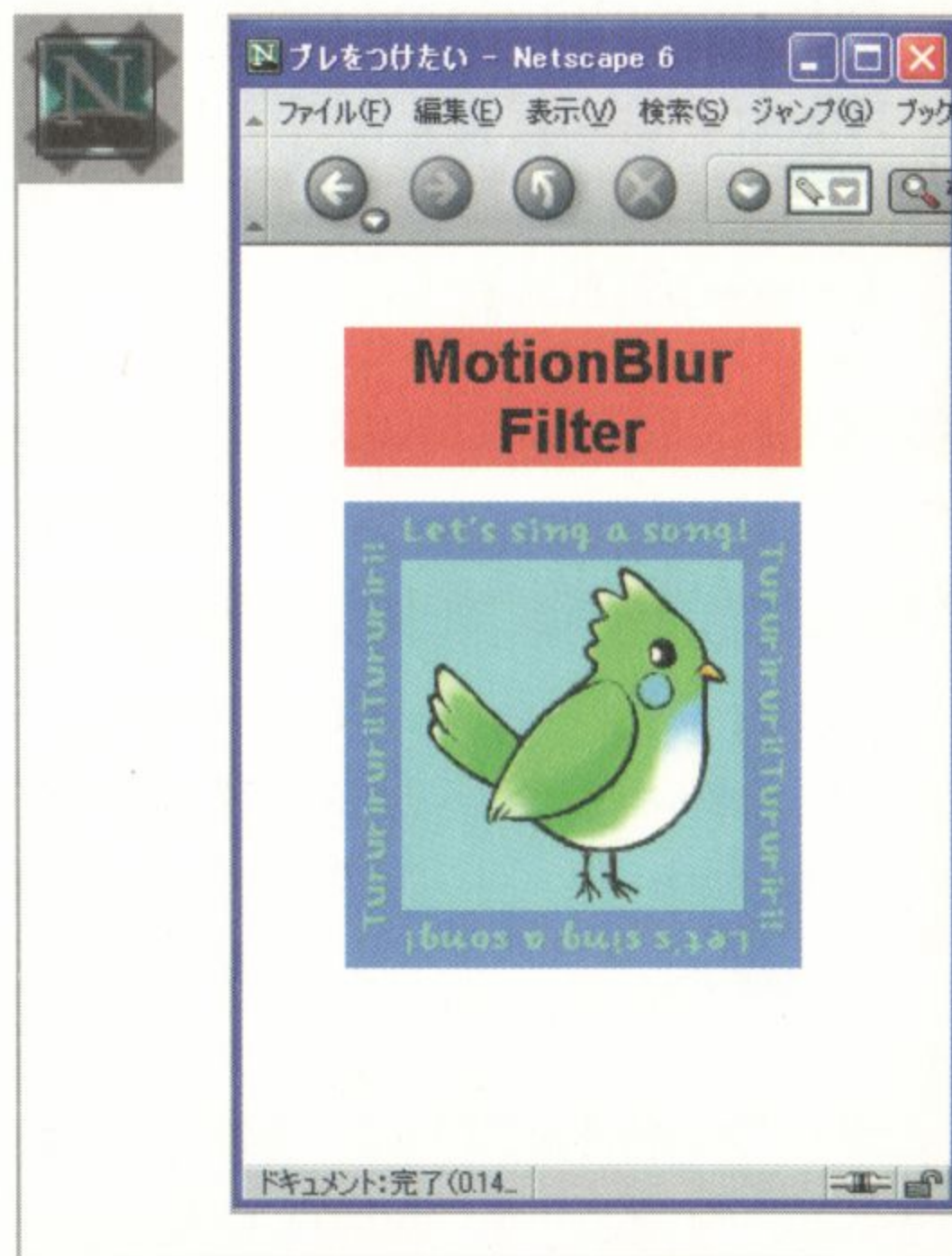
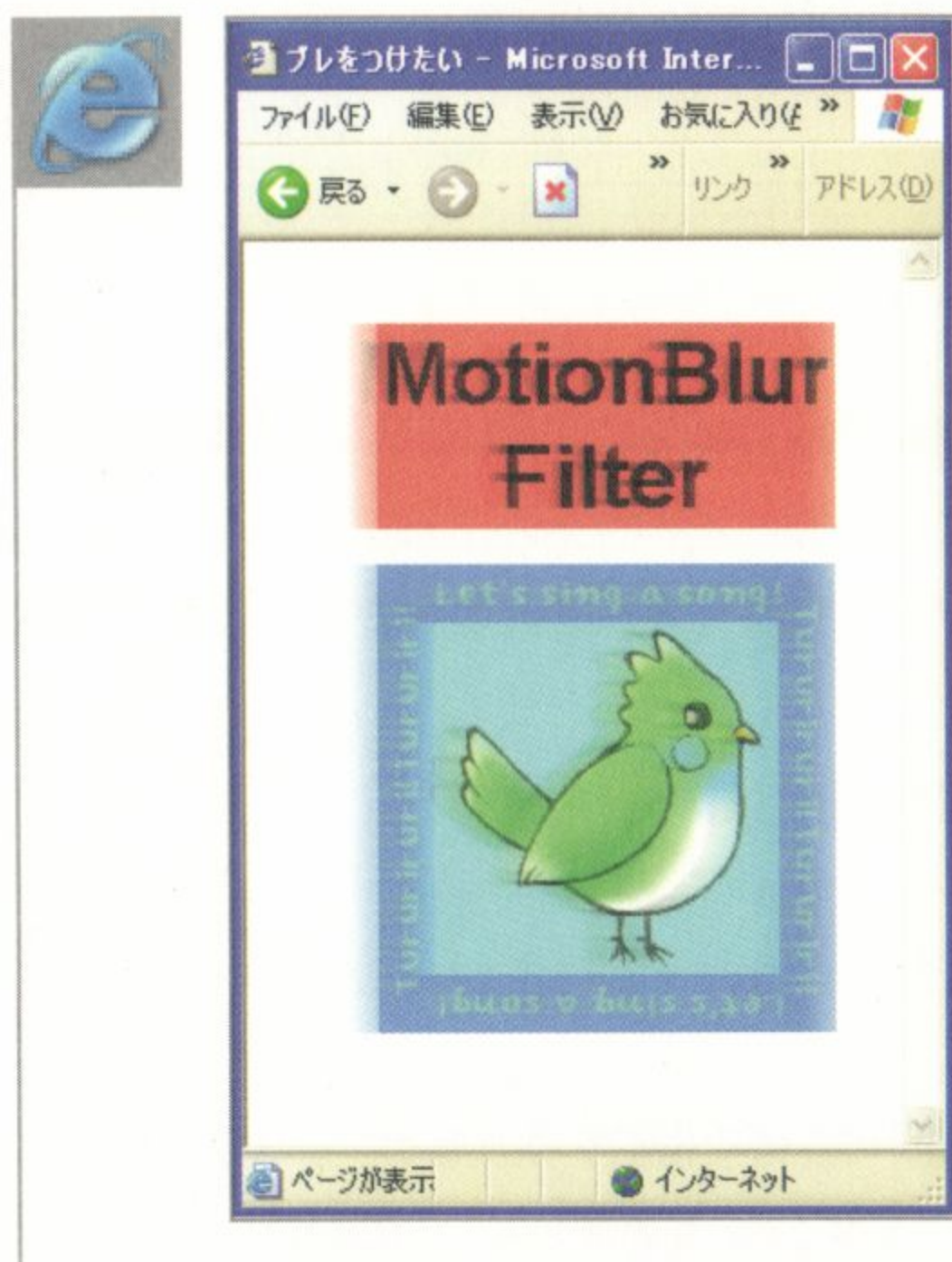
strengthは、ブレの強さを0以上の整数値で指定します。数値が大きいほどブレが強くなり、ブレも長くなります。

addでは、ブレを入れた画像を元のスタイルの上に表示するかどうかをtrueまたはfalseで指定します。trueで元のスタイルに上書きし、falseで上書きしません。デフォルトはfalseで、この場合はフィルタ効果の結果のみ表示されます。

enabledはフィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>ブレをつけたい</title>
<style type="text/css">
<!--
.sample		{
	filter:progid:DXImageTransform.Microsoft.MotionBlur(direction=270,
	strength=15,add=true)
}
body		{ margin: 0 }
div		{
	width: 250px;
	height: 280px;
	text-align: center;
	position: absolute;
	top: 20px;
	left: 20px
}
p		{
	width: 200px;
	margin: 15px auto;
	font: bold 1.3em Arial,Helvetica,sans-serif;
	background-color: #ff6633
}
-->
</style>
</head>
<body>
<div class="sample">
<p>MotionBlur Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
```

```
</div>
</body>
</html>
```

IE4.0以上のブラウザでの指定方法

IE4.0以上のブラウザで同様のフィルタを表現する場合は次のように指定します。設定方法はMotionBlurを指定した場合と同じです。

なお、Blurフィルタは IE4.0と5.5以上では性質が異なりますので注意してください（p.276参照）。

```
.sample {
        filter:blur(direction=270,strength=15,add=true)
}
```

11 FILTER

影を伸ばしたい

filter:progid:DXImageTransform.Microsoft.Shadow(★) 【IE5.5以上】
filter:shadow(★) 【IE4.0以上】

★••••••プロパティ

影をこすったように伸ばします。

指定できるプロパティは次の通りです。

color	影の色
direction	影の伸びる方向（45度単位）
strength	影の長さ（1～255）
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

colorで影の色を指定します。色の指定には、RGBの数値で指定する方法と、キーワードで指定する方法とがあります。色の詳しい指定方法についてはp.47を参照してください。

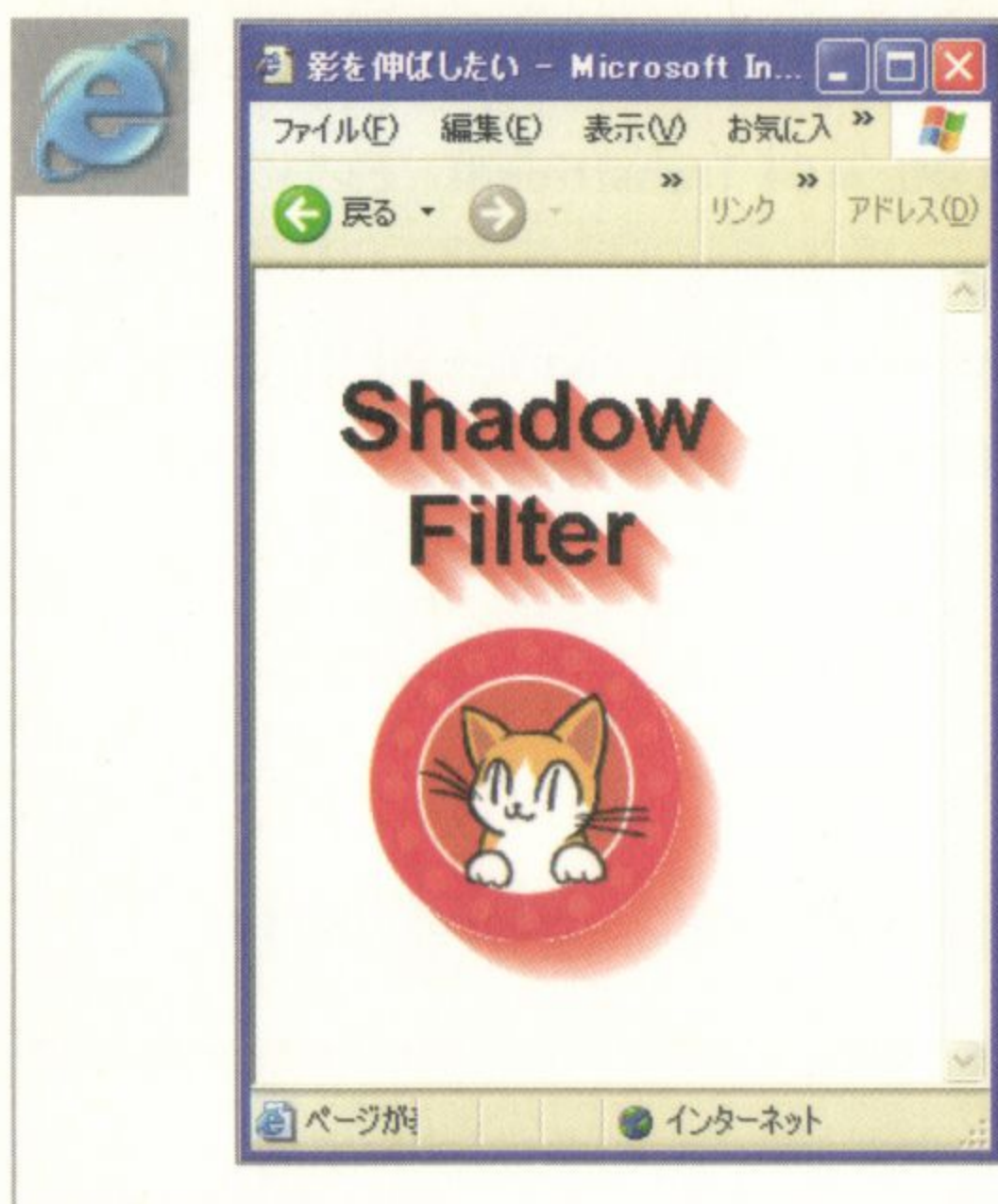
directionは影を伸ばす方向を角度で指定します。たとえば、0で真上へ、180で真下の方向となります。

directionの値	0	45	90	135	180	225	270	315
影の方向	上	右上	右	右下	下	左下	左	左上

strengthは影の長さを1～255から指定します。デフォルトの値は5です。

enabledはフィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>影を伸ばしたい</title>
<style type="text/css">
<!--
.sample        {
        filter:progid:DXImageTransform.Microsoft.Shadow(color=#ff6633,
        direction=135,strength=15)
}
body           { margin: 0 }
div            {
        width: 170px;
        height: 220px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p              {
        width: 140px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Shadow Filter</p>
<img src="cat.gif" alt="" width="120" height="120">
</div>
</body></html>
```

フィルタ

IE4.0以上のブラウザでの指定方法

IE4.0以上のブラウザで同様のフィルタを表現する場合は次のように指定します。設定方法はShadowを指定した場合と同じです。

```
.sample {
        filter:shadow(color=#ff6633,direction="135",strength=15)
}
```

12 FILTER

ウェーブをかけたい

filter:progid:DXImageTransform.Microsoft.Wave(★) 【IE5.5以上】

filter:wave(★) 【IE4.0以上】

★……プロパティ

波打ったような効果を表現します。

指定できるプロパティは次の通りです。

freq	波の数
lightstrength	光の強さ（0～100）
phase	波の開始位置（0～100）
strength	波の振幅
add	元の画像を合成するかどうか（true、falseもしくは1、0）
enabled	フィルタを実行するかどうか（true、falseもしくは1、0）

freqは波の数を整数値で指定します。指定する値が大きいほど波の数が多くなり、ウェーブの間隔が狭くなります。

lightstrengthはウェーブにあたる光の強さを0から100までの整数値で指定します。0では光がまったくあたらない状態となり、ウェーブの効果は要素の輪郭以外に現れません。100では光がもっとも強くあたっている状態となり、ウェーブの谷の部分は黒になります。

phaseはウェーブの始まる位置を0から100までの整数値で指定します。

strengthはウェーブの振幅（横方向へのゆれ）を指定します。数値が大きいほどゆれも大きくなります。デフォルトの値は5です。

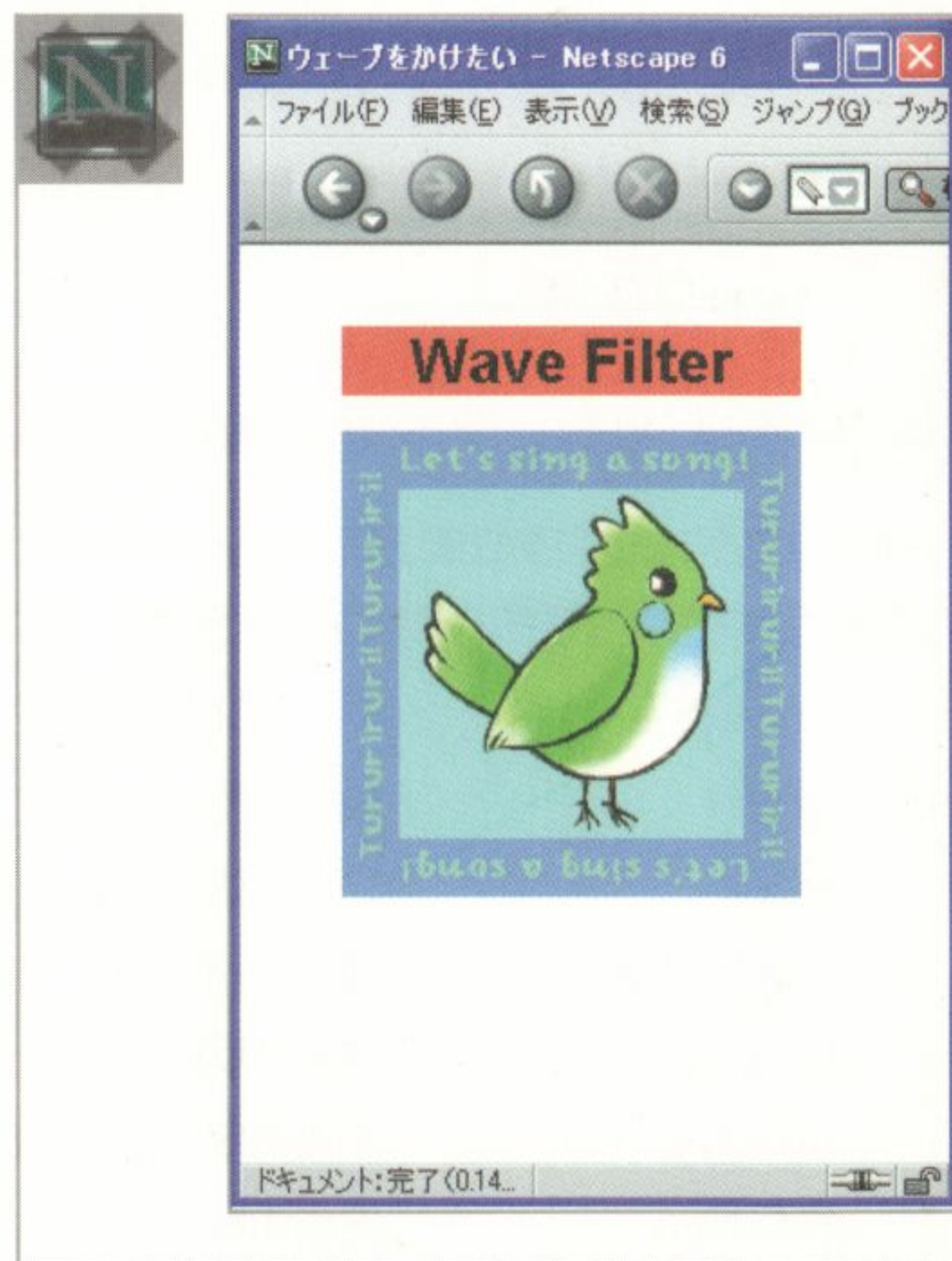
addは元の画像を合成するかどうかを指定します。trueまたは1を指定すると元の画像を合成し、falseまたは0を指定すると元の画像を合成しません。

enabledでは、フィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>ウェーブをかけたい</title>
<style type="text/css">
<!--
.sample        {
        filter:progid:DXImageTransform.Microsoft.Wave(freq=4,
        lightstrength=20,phase=30,strength=10,add=false)
}
body           { margin: 0 }
div            {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p              {
        width: 200px;
        margin: 15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Wave Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
```

フィルタ

```
</div>
</body>
</html>
```

IE4.0以上のブラウザでの指定方法

IE4.0以上のブラウザで同様のフィルタを表現する場合は次のように指定します。設定方法はWaveを指定した場合と同じです。

```
.sample {
        filter:wave(freq=4,lightstrength=20,phase=30,strength=10,add=false)
}
```

フィルタ

13 FILTER

さまざまな効果をまとめて設定したい

filter:progid:DXImageTransform.Microsoft.BasicImage(★) 【IE5.5 以上】

★……プロパティ

画像や文字に対するさまざまなフィルタ効果をまとめて設定します。

指定できるプロパティは次の通りです。

grayscale	グレースケール化（1、0）
invert	色の反転（1、0）
mask	マスク処理（1、0）
maskcolor	塗りつぶす色
mirror	左右の反転（1、0）
opacity	不透明度（0.0～1.0）
rotation	回転角度（0～3）
xray	白黒反転（1、0）
enabled	フィルタを実行するかどうか（true、false もしくは 1、0）

grayscale はグレースケールにするかどうかを、invert は色を反転して表示するかどうかを指定します。どちらもデフォルトの値は 0（通常の色で表示する）です。

mask はマスク処理をして表示するかどうかを指定します。マスク処理をする（1）よう指定すると、文字や画像の不透明部分を切り抜いて透明にし、背景や透過処理をした部分など本来透明な部分を maskcolor プロパティで指定した色で塗りつぶします。デフォルトは 0（マスク処理をしない）です。

maskcolor はマスク処理をして塗りつぶす色を RGB 形式で指定します。

mirror は左右を反転するかどうかを指定します。デフォルトの値は 0（反転しない）です。

opacity は不透明度を、透明の 0.0 から不透明の 1.0 までの範囲で整数値で指定します。デフォルトの値は 1.0（不透明）です。

rotation は回転して表示させる場合の角度を次の値から指定します。デフォルトは 0 で回転しません。

rotationの値	0	1	2	3
回転の角度	回転しない	90度	180度	270度

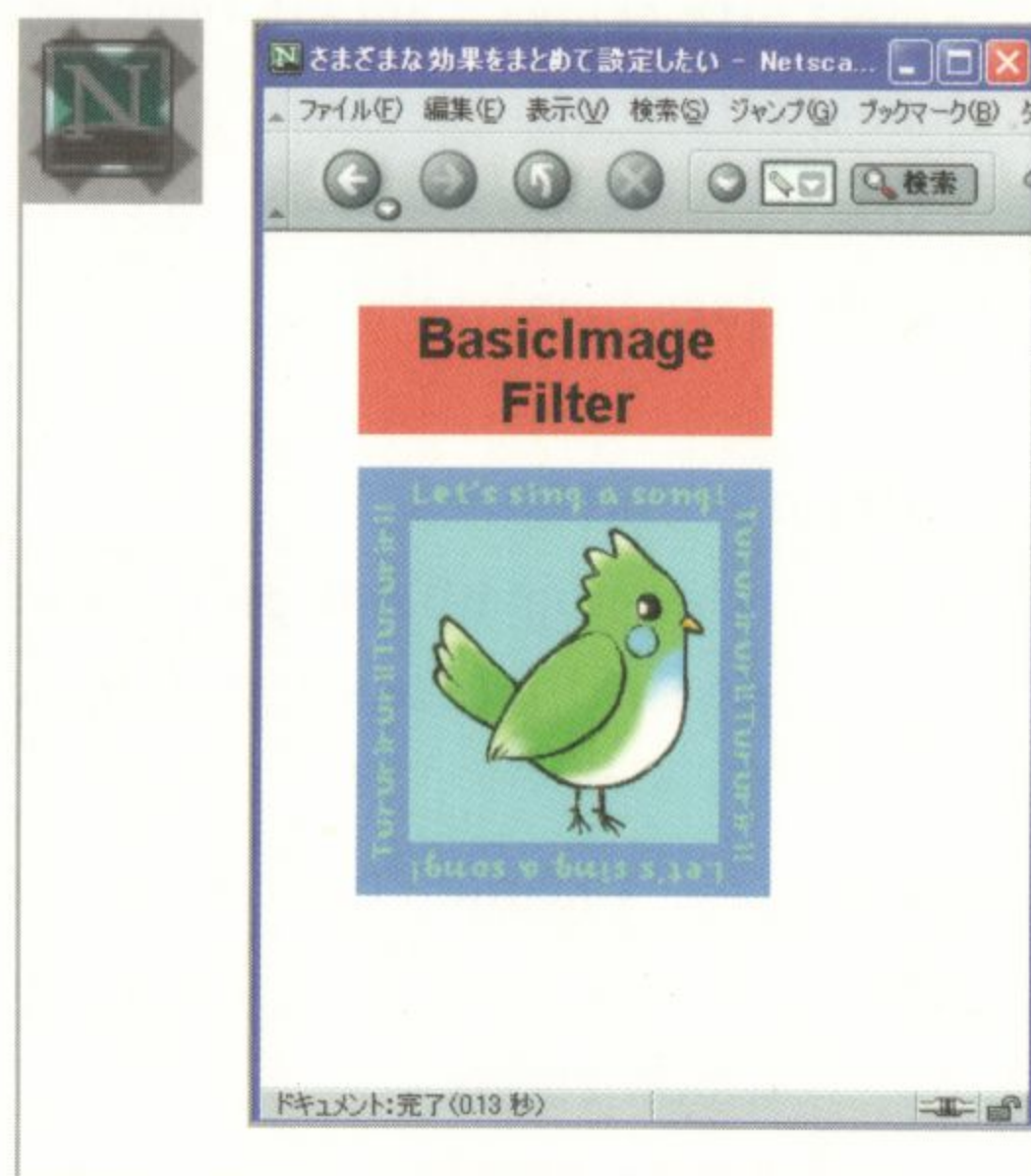
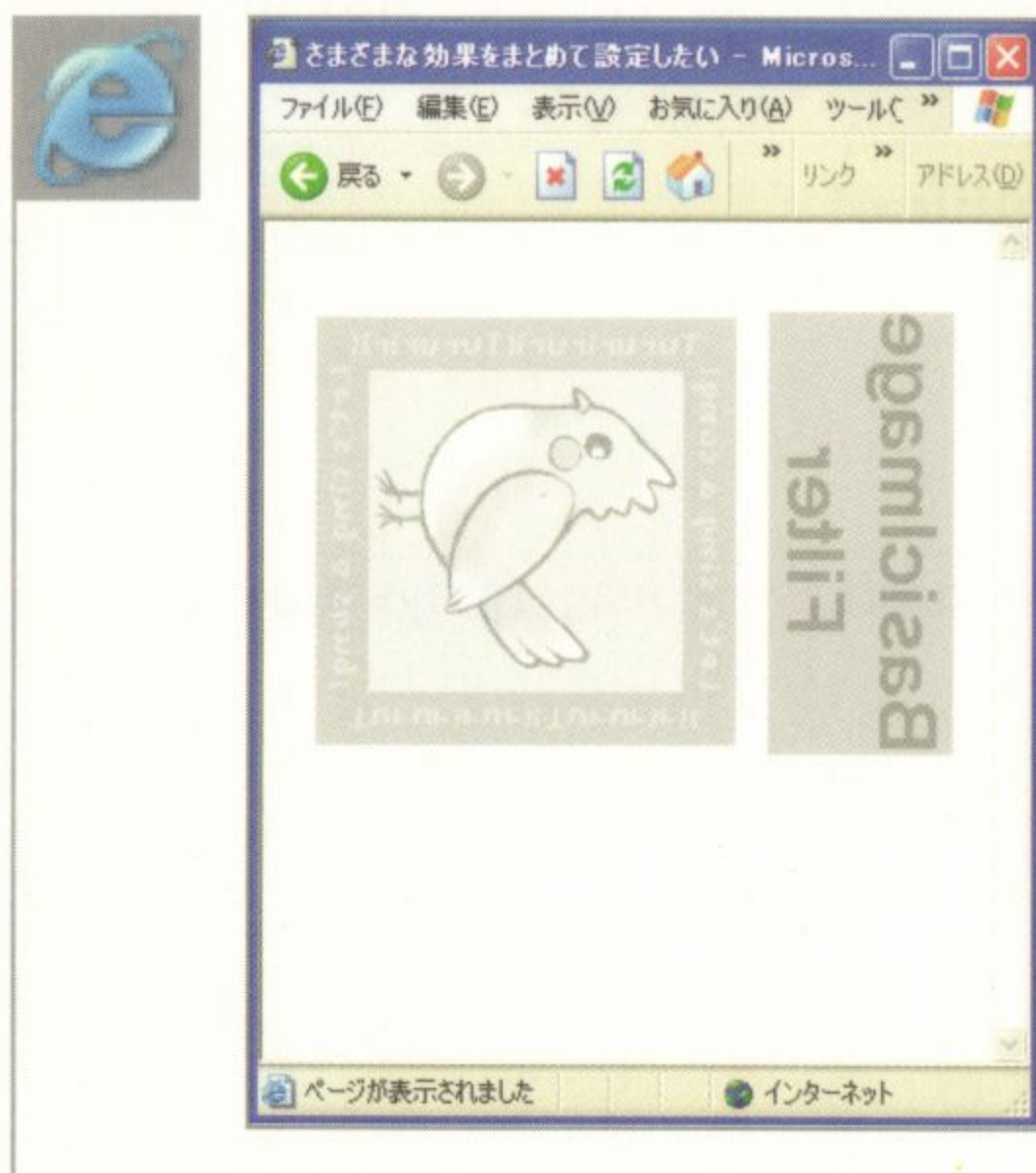
xrayはX線写真のように白黒反転をして表示するかどうかを指定します。デフォルトの値は0（通常の色で表示する）です

enabledはフィルタを実行するかどうかを指定します。trueまたは1を指定するとフィルタを実行し、falseまたは0を指定するとフィルタを実行しません。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>さまざまな効果をまとめて設定したい</title>
<style type="text/css">
<!--
.sample        {
        filter:progid:DXImageTransform.Microsoft.BasicImage(grayscale=1,
        mirror=1,opacity=0.35,rotation=3)
}
body          { margin: 0 }
div           {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p             {
        width: 200px;
        margin:15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
-->
</style>
```

フィルタ

```
</head>
<body>
<div class="sample">
<p>BasicImage Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body>
</html>
```

14 FILTER

表示方向を反転させたい

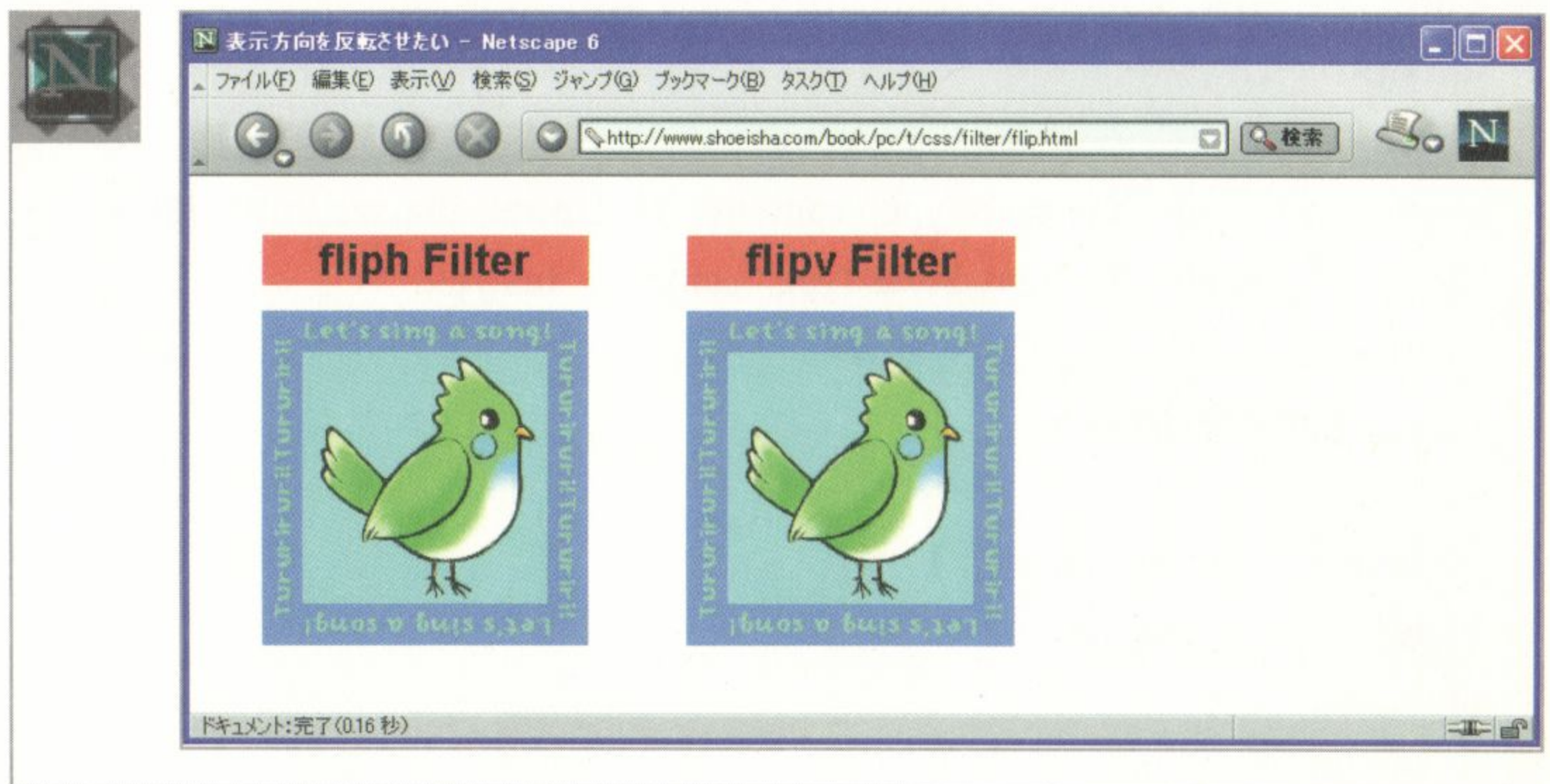
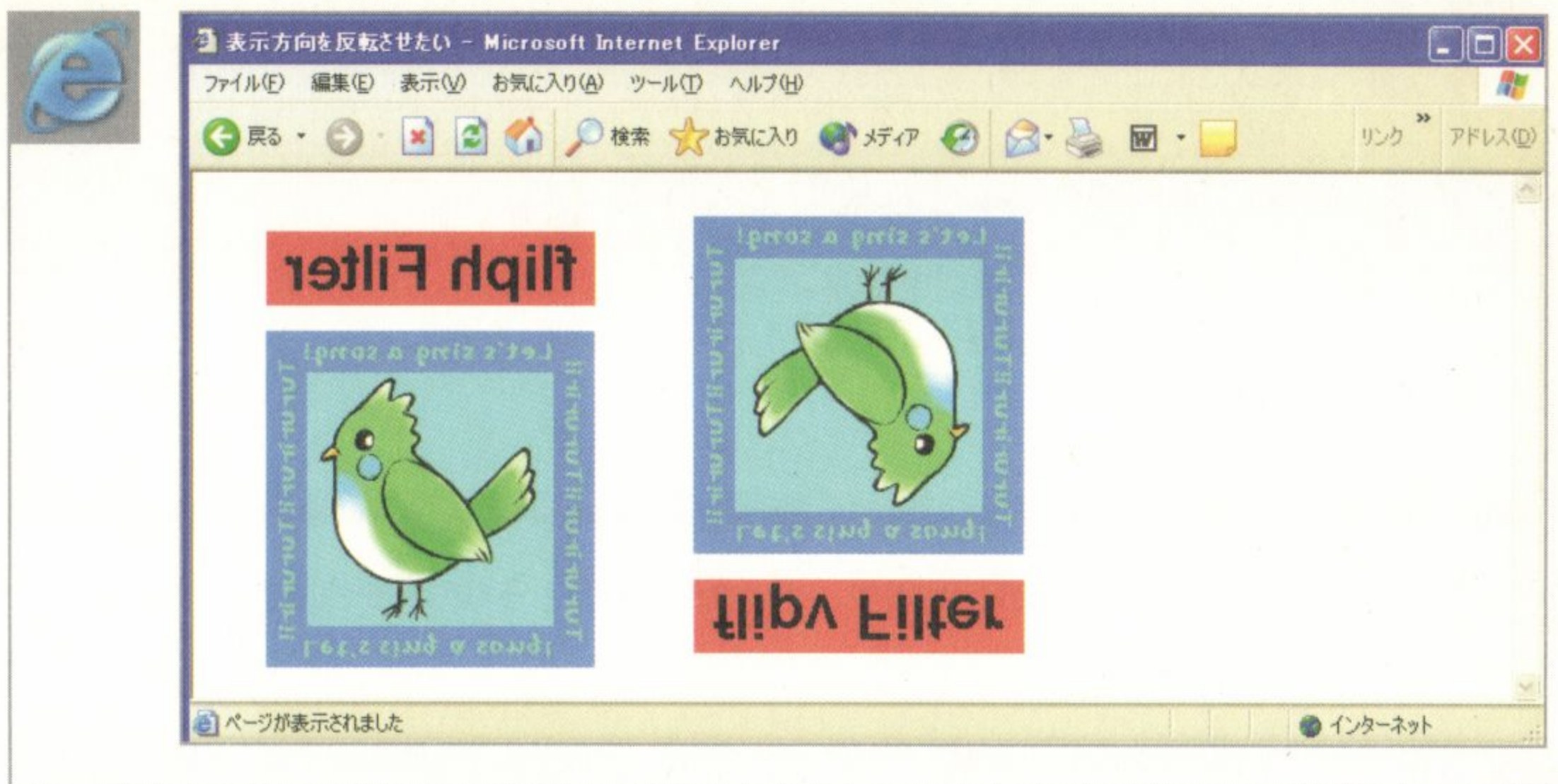
filter:fliph()	左右の反転	【IE4.0以上】
filter:flipv()	上下の反転	【IE4.0以上】

文字や画像の表示方向を反転させて表示します。
fliphは対象を左右に、flipvは上下に反転させます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>表示方向を反転させたい</title>
<style type="text/css">
<!--
.sample1        { filter: fliph() }
.sample2        { filter: flipv() }
body            { margin: 0 }
div#flipsh      {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
div#flipsv      {
        width: 250px;
        height: 280px;
        text-align: center;
```

フィルタ

```
        position: absolute;
        top: 20px;
        left: 280px
}
p               {
        width: 200px;
        margin:15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
-->
</style>
</head>

<body>
<div class="sample1" id="flipsh">
<p>flipsh Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
<div class="sample2" id="flipsv">
<p>fliphv Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body>
</html>
```

フィルタ

フィルタ

IE5.5以上のブラウザでの指定方法

IE5.5以上のブラウザで同様のフィルタを表現する場合は次のように指定します。詳しくはBasicImageの項（p.304）を参照してください。

```
.sample1 {
        filter:progid:DXImageTransform.Microsoft.BasicImage(mirror=1)
}
.sample2 {
        filter:progid:DXImageTransform.Microsoft.BasicImage(rotation=2, mirror=1)
}
```

15 FILTER

グレースケールにしたい

filter:gray() 【IE4.0以上】

文字や画像をグレースケールで表示します。モノクロ写真のような効果を表現できます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>グレースケールにしたい</title>
<style type="text/css">
<!--
.sample         { filter: gray() }
body            { margin: 0 }
div             {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p               {
        width: 200px;
        margin:15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
```

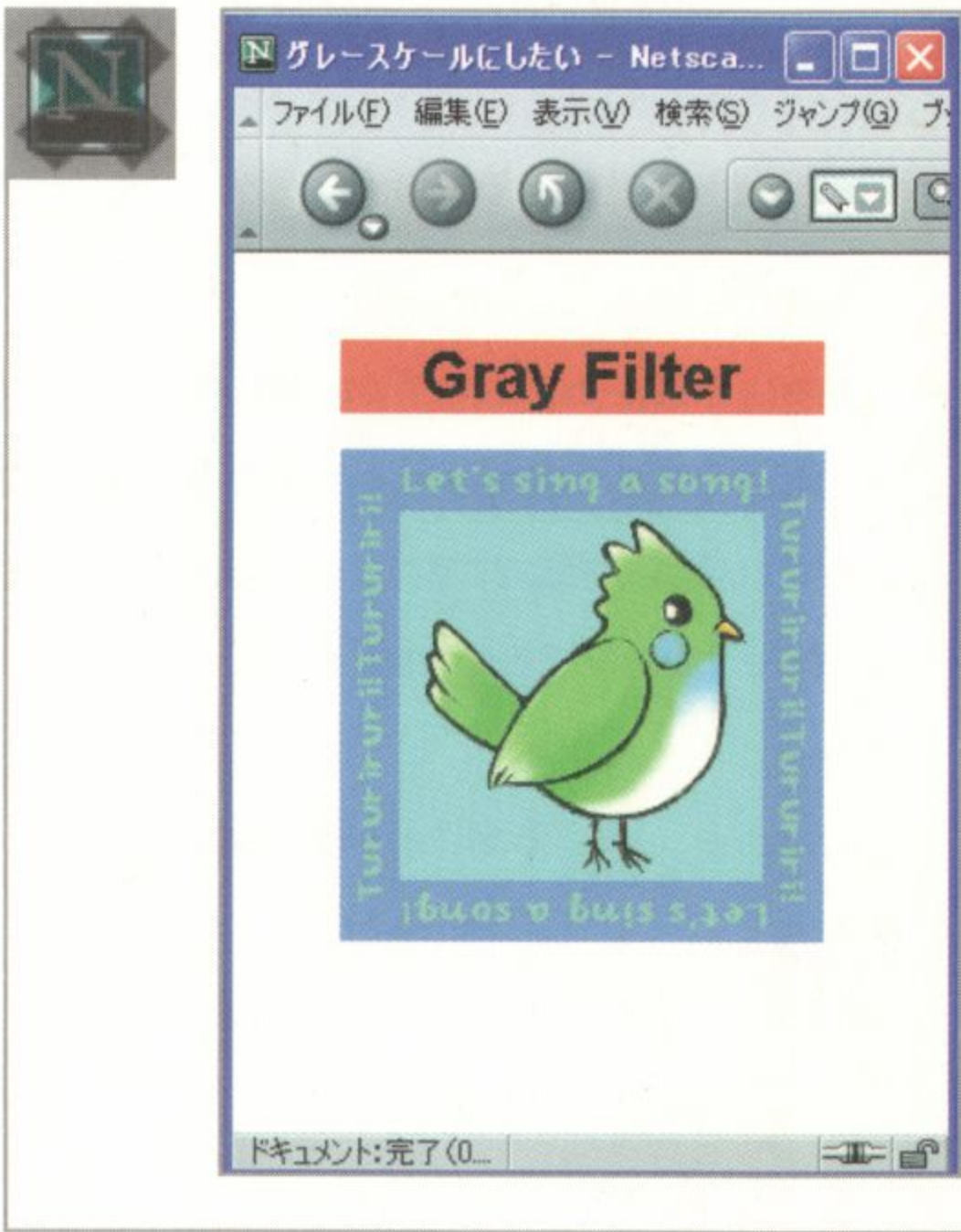
フィルタ

```
-->
</style>
</head>
<body>
<div class="sample">
<p>Gray Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body>
</html>
```

フィルタ

IE5.5以上のブラウザでの指定方法

IE5.5以上のブラウザで同様のフィルタを表現する場合は次のように指定します。詳しくはBasicImageの項（p.304）を参照してください。

```
.sample {
        filter:progid:DXImageTransform.Microsoft.BasicImage(grayscale=1)
}
```

16 FILTER

色を反転させたい

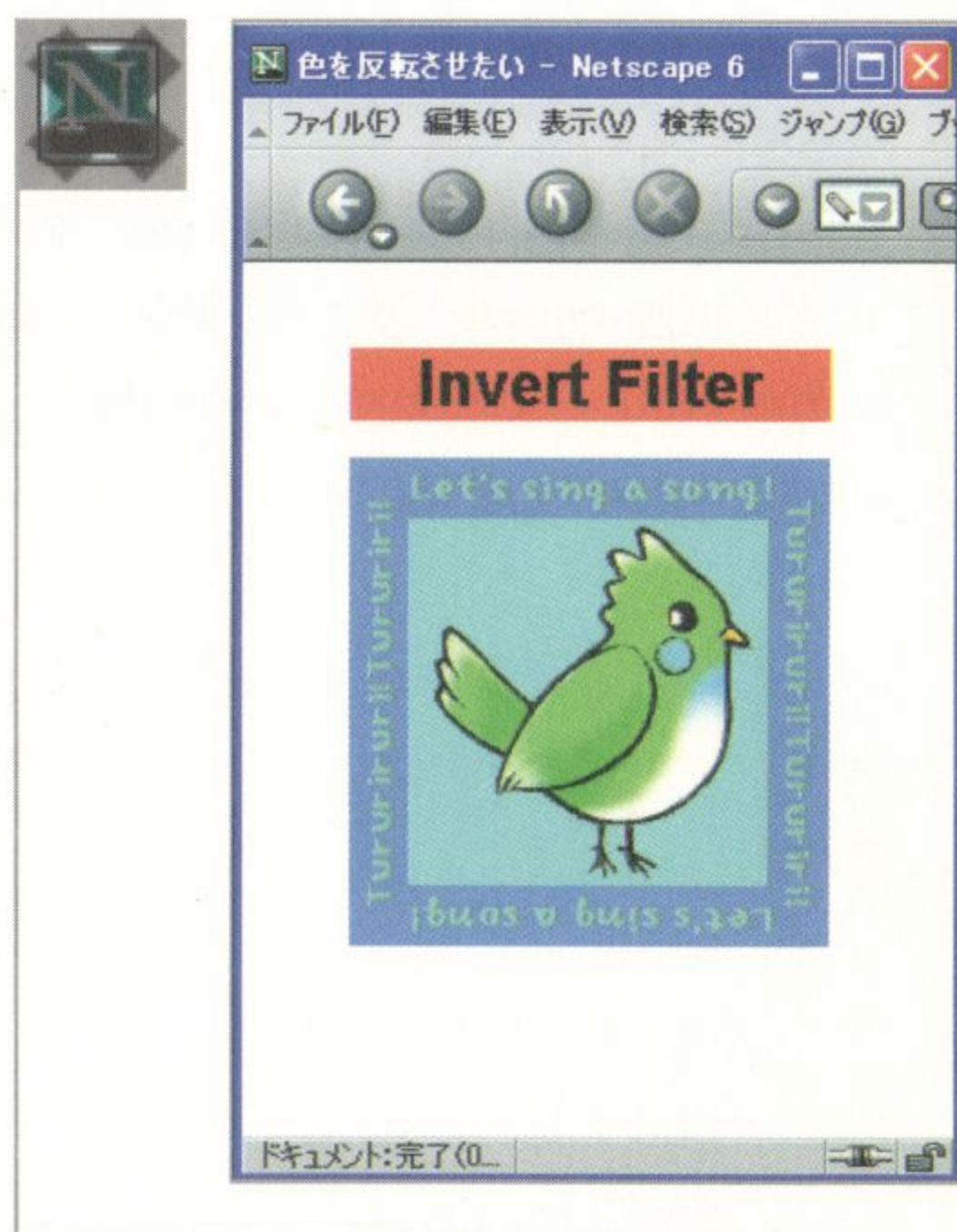
filter:invert()　【IE4.0以上】

文字や画像の色を反転して表示します。
白色は黒色へ、黄色は青色へ、赤色は水色へといったように処理が行われます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>色を反転させたい</title>
<style type="text/css">
<!--
.sample         { filter: invert() }
body            { margin: 0 }
div             {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p               {
        width: 200px;
        margin:15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
```

フィルタ

```
}
-->
</style>
</head>
<body>
<div class="sample">
<p>Invert Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body>
</html>
```

IE5.5以上のブラウザでの指定方法

IE5.5以上のブラウザで同様のフィルタを表現する場合は次のように指定します。詳しくはBasicImageの項（p.304）を参照してください。

```
.sample {
        filter:progid:DXImageTransform.Microsoft.BasicImage(invert=1)
}
```

フィルタ

17 FILTER

X線フィルタをかけたい

filter:xray() 【IE4.0以上】

画像をX線写真のように表示させます。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>X線フィルタをかけたい</title>
<style type="text/css">
<!--
.sample        { filter:xray() }
body           { margin: 0 }
div            {
        width: 250px;
        height: 280px;
        text-align: center;
        position: absolute;
        top: 20px;
        left: 20px
}
p              {
        width: 200px;
        margin:15px auto;
        font: bold 1.3em Arial,Helvetica,sans-serif;
        background-color: #ff6633
}
```

フィルタ

```
-->
</style>
</head>
<body>
<div class="sample">
<p>Xray Filter</p>
<img src="bird.jpg" alt="" width="200" height="200">
</div>
</body>
</html>
```

フィルタ

IE5.5以上のブラウザでの指定方法

IE5.5以上のブラウザで同様のフィルタを表現する場合は次のように指定します。詳しくはBasicImageの項（p.304）を参照してください。

```
.sample {
        filter:progid:DXImageTransform.Microsoft.BasicImage(xray=1)
}
```

1 OTHER

カーソルの形状を指定したい

cursor: ★

★……キーワード

マウスなどのポインティングデバイスの位置を示すカーソルの形状を指定します。
値には以下のキーワードがあります。

キーワード

キーワード	説明	IE6	N6.2
auto	ブラウザが自動的に指定（デフォルト）		
crosshair	十字型	＋	＋
default	標準的なカーソル。矢印が多い		
pointer	アンカー上にあることを示す		
move	対象が移動可能であることを示す		
e-resize	右方向にリサイズ可能であることを示す	⇨	↔
ne-resize	右上方向にリサイズ可能であることを示す		⤢
nw-resize	左上方向にリサイズ可能であることを示す		⤡
n-resize	上方向にリサイズ可能であることを示す	⇧	↕
se-resize	右下方向にリサイズ可能であることを示す		⤡
sw-resize	左下方向にリサイズ可能であることを示す		⤢
s-resize	下方向にリサイズ可能であることを示す	⇩	↕
w-resize	左方向にリサイズ可能であることを示す	⇦	↔
text	文字を範囲指定できることを示す	I	I
wait	処理中であることを示す		
help	ヘルプを利用できることを示す	?	?

Internet Explorer 6 から追加されたキーワード

キーワード	説明	IE6	N6.2
all-scroll	ページが上下左右にスクロール可能であることを示す		未対応
col-resize	左右にリサイズ可能であることを示す		未対応
no-drop	ドラッグした対象がドロップできない領域であることを示す		未対応
not-allowed	要求されたアクションが動作できないことを示す		未対応
progress	バックグラウンドで処理が行われていることを示す		未対応

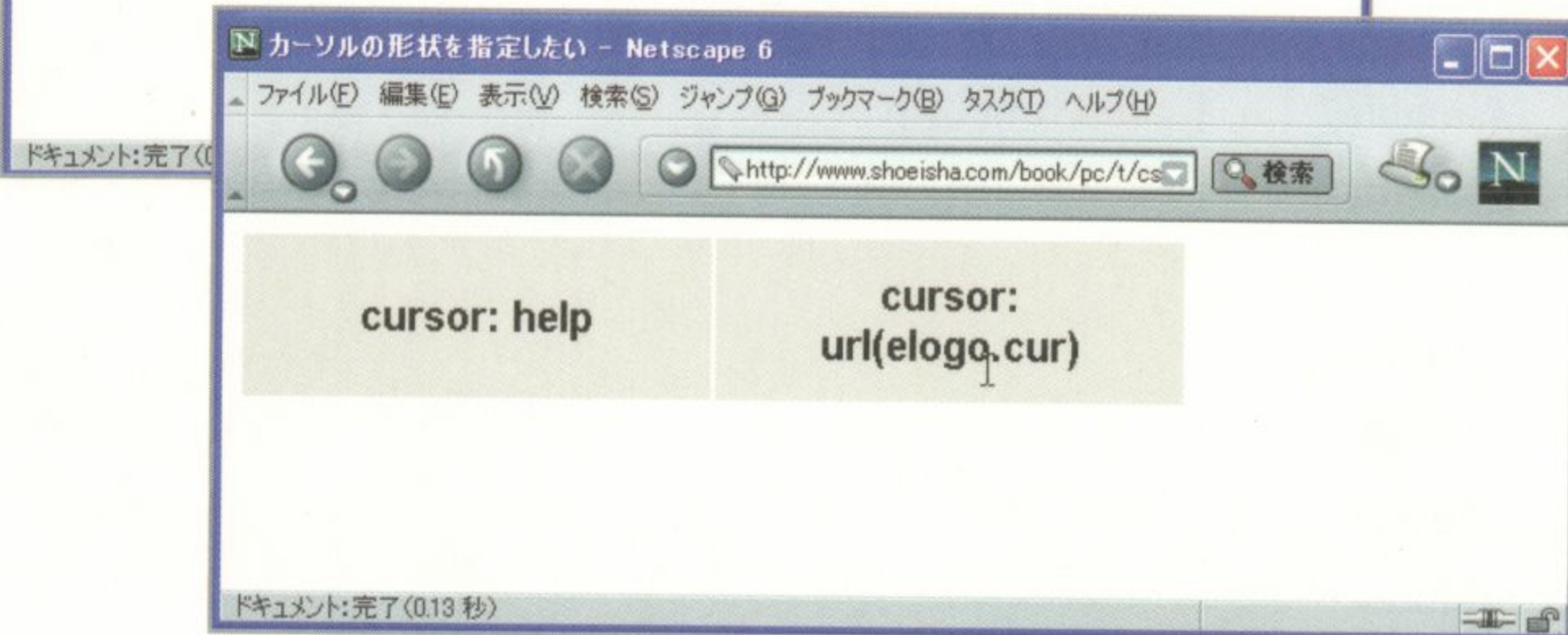
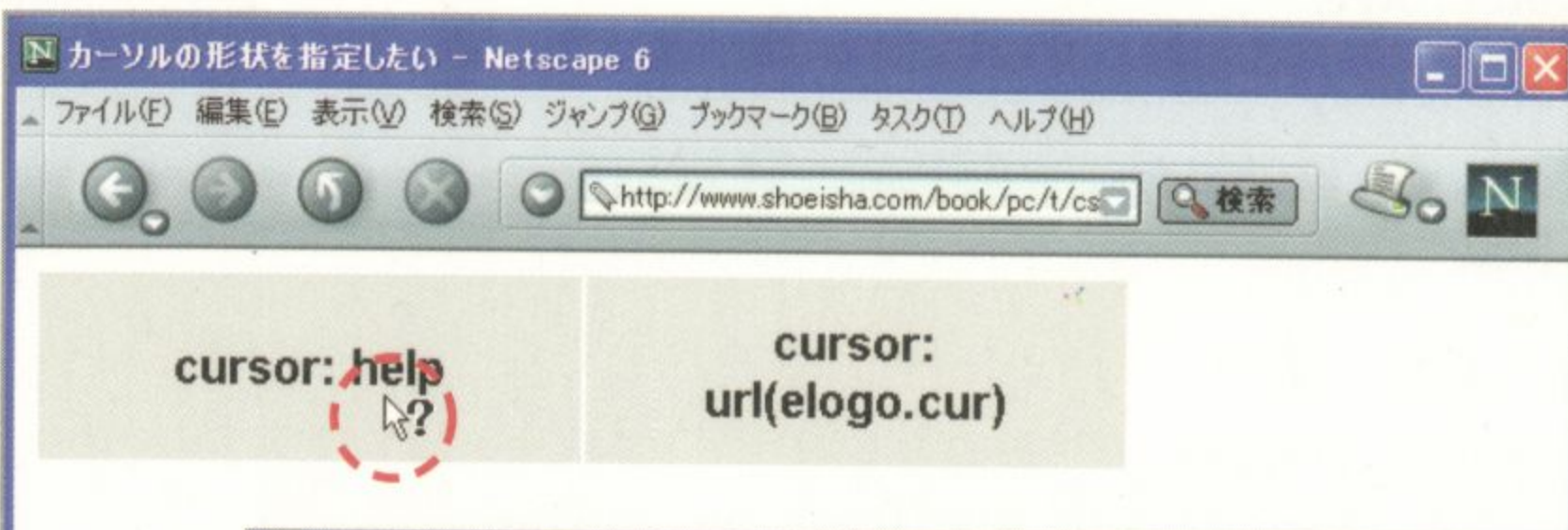
その他

row-resize	上下にリサイズ可能であることを示す	≑	未対応
url(☆)	任意のカーソル（拡張子.curと.aniのみ。☆にURLを指定）		未対応
vertical-text	縦書き文字を範囲指定できることを示す	⊢⊣	未対応

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>カーソルの形状を指定したい</title>
<style type="text/css">
<!--
td              {
        background-color: #e4e4e4;
        color: black;
        font-family: Arial,Helvetica,sans-serif;
        font-weight: bold;
        width: 200px;
        padding: 15px;
        text-align: center
}
#sample1        { cursor: help }
#sample2        { cursor: url(elogo.cur) }
-->
</style>
</head>
<body>
<table>
<tr>
  <td id="sample1">cursor: help</td>
  <td id="sample2">cursor: url(elogo.cur)</td>
</tr>
</table>
</body>
</html>
```

その他

▲Internet Explorer 6では拡張子.cur、.aniの画像をカーソルに指定することもできます

▲Netscape 6.2は画像をカーソルに指定できるurl()には対応していません

その他

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
auto	○	○	○	○	×	×	○
crosshair	○	○	○	○	×	×	○
default	○	○	○	○	×	×	○
pointer	×	×	×	○	×	×	○
move	○	○	○	○	×	×	○
e-resize	○	○	○	○	×	×	○
ne-resize	○	○	○	○	×	×	○
nw-resize	○	○	○	○	×	×	○
n-resize	○	○	○	○	×	×	○
se-resize	○	○	○	○	×	×	○
sw-resize	○	○	○	○	×	×	○
s-resize	○	○	○	○	×	×	○
w-resize	○	○	○	○	×	×	○
text	○	○	○	○	×	×	○
wait	○	○	○	○	×	×	○
help	○	○	○	○	×	×	○
all-scroll	×	×	×	○	×	×	×
col-resize	×	×	×	○	×	×	×
no-drop	×	×	×	○	×	×	×
not-allowed	×	×	×	○	×	×	×
progress	×	×	×	○	×	×	×
row-resize	×	×	×	○	×	×	×
url()	×	×	×	○	×	×	×
vertical-text	×	×	×	○	×	×	×

※適用するセレクタによっても効果が変わります
※ Macintosh 版 IE5 は pointer にも対応しています

その他

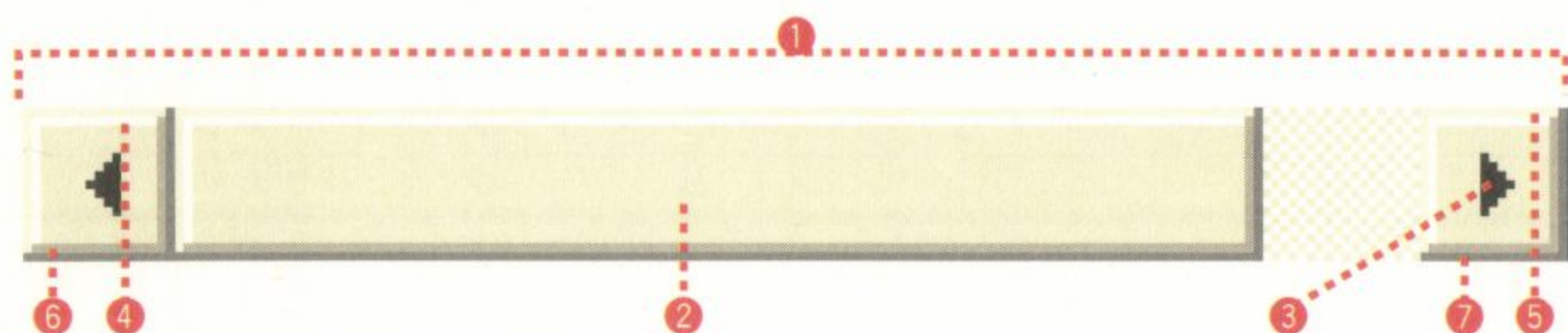
2 OTHER

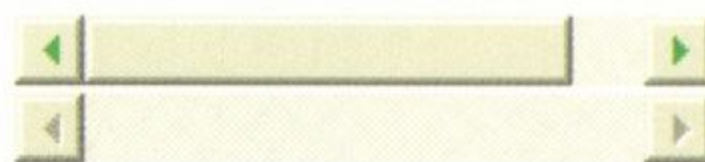
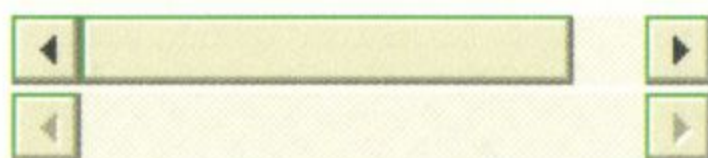
スクロールバーの色を設定したい

scrollbar-base-color: ★	基本の色
scrollbar-face-color: ★	表面の色
scrollbar-arrow-color: ★	矢印の色
scrollbar-highlight-color: ★	ハイライト部分の色
scrollbar-3dlight-color: ★	ボタンのハイライト部分の色
scrollbar-shadow-color: ★	シャドウ部分の色
scrollbar-darkshadow-color: ★	ボタンのシャドウ部分の色

★……キーワード
色指定値

スクロールバーの色を指定します。Internet Explorerが独自に拡張したプロパティで、IE 5.5以上で有効になります。なお、このプロパティを指定するとスクロールバーの形状がWindows Meまでのものに変更されます。

各プロパティが制御するのは下図の部分です。

❶ base	全体の基本となる色	❷ face	スクロールバーの表面の色
❸ arrow	矢印の部分の色	❹ highlight	ボタン内側の左と上の枠色
❺ 3dlight	ボタン外側の左と上の枠色	❻ shadow	ボタン内側の右と下の枠色
❼ darkshadow	ボタン外側の右と下の枠色		

▲scrollbar-base-color ▲scrollbar-face-color ▲scrollbar-arrow-color

▲scrollbar-highlight-color ▲scrollbar-3dlight-color

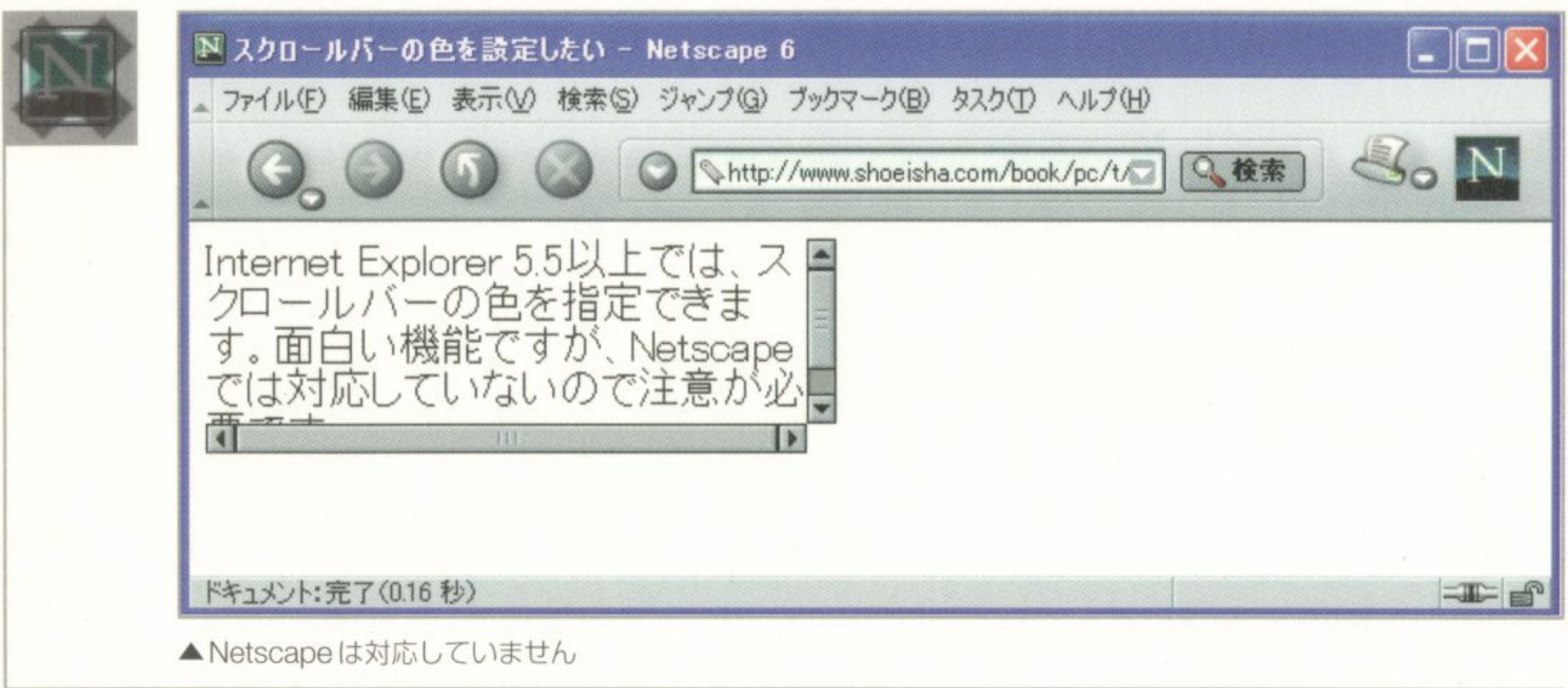
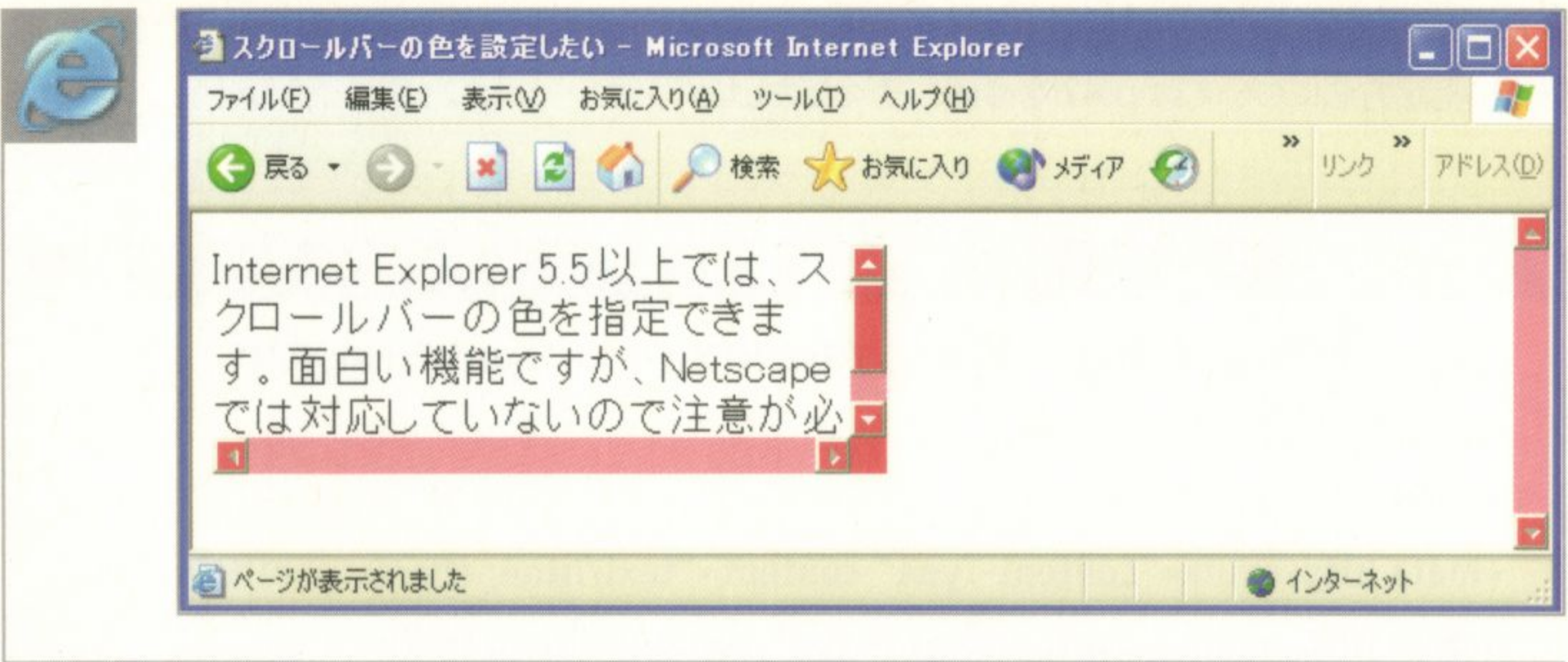
▲scrollbar-shadow-color ▲scrollbar-darkshadow-color

色の指定には、RGBの数値で指定する方法と、キーワードで指定する方法とがあります。色の詳しい指定方法についてはp.47を参照してください。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>スクロールバーの色を設定したい</title>
<style type="text/css">
<!--
body,div        {
        scrollbar-arrow-color: #ffffff;
        scrollbar-face-color: #ff0000;
        scrollbar-track-color: #ff9999
}
div             {
        width: 15em;
        height: 5em;
        overflow: scroll
}
-->
</style>
</head>
<body>
<div>Internet Explorer 5.5以上では、スクロールバーの色を指定できます。面白い機能ですが、Netscapeでは対応していないので注意が必要です。</div>
</body>
</html>
```

その他

▲Netscapeは対応していません

共通	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
キーワード	×	×	○	○	×	×	×
色指定値	×	×	○	○	×	×	×

※適用するセレクタによっても効果が変わります

参照
内容領域の幅と高さを指定したい……………p.198
内容があふれる場合の処理方法を指定したい……p.230
内容があふれる場合の横方向の処理方法を指定したい……p.238
内容があふれる場合の縦方向の処理方法を指定したい……p.234

その他

3 OTHER

IMEの入力状態を指定したい

ime-mode: ★

★……キーワード

日本語入力システムのON/OFFを設定します。Internet Explorerが独自に拡張したプロパティで、IE 5以降で有効になります。

値には以下のキーワードがあります。

auto	IMEに作用しない。このプロパティを指定しない場合と同様（デフォルト）
active	IMEを有効にする（オン）
inactive	IMEを無効にする（オフ）
disabled	IMEを完全に無効にする（オフ）

activeで日本語入力をオンにし、inactiveでオフにしますが、いずれもユーザーの操作で変更は可能です。disactiveは完全にIMEを使用できないようにするためユーザーが日本語入力をオンにすることもできません。

このプロパティを指定することで、ユーザーがIMEのON/OFFを切り替える手間を省いたり、全角半角の入力ミスを防いだりすることが可能になり、入力時の操作性が高まります。

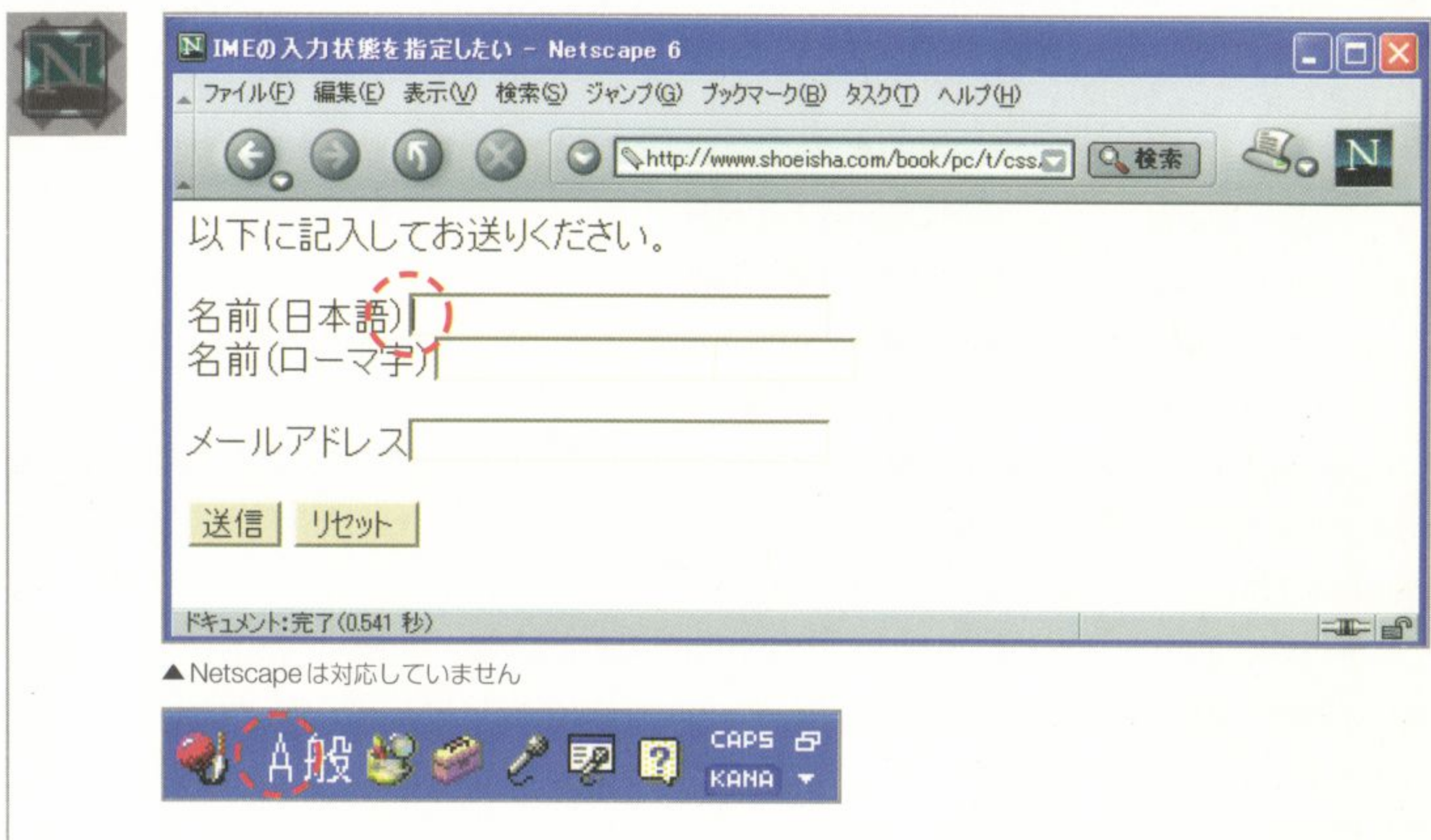
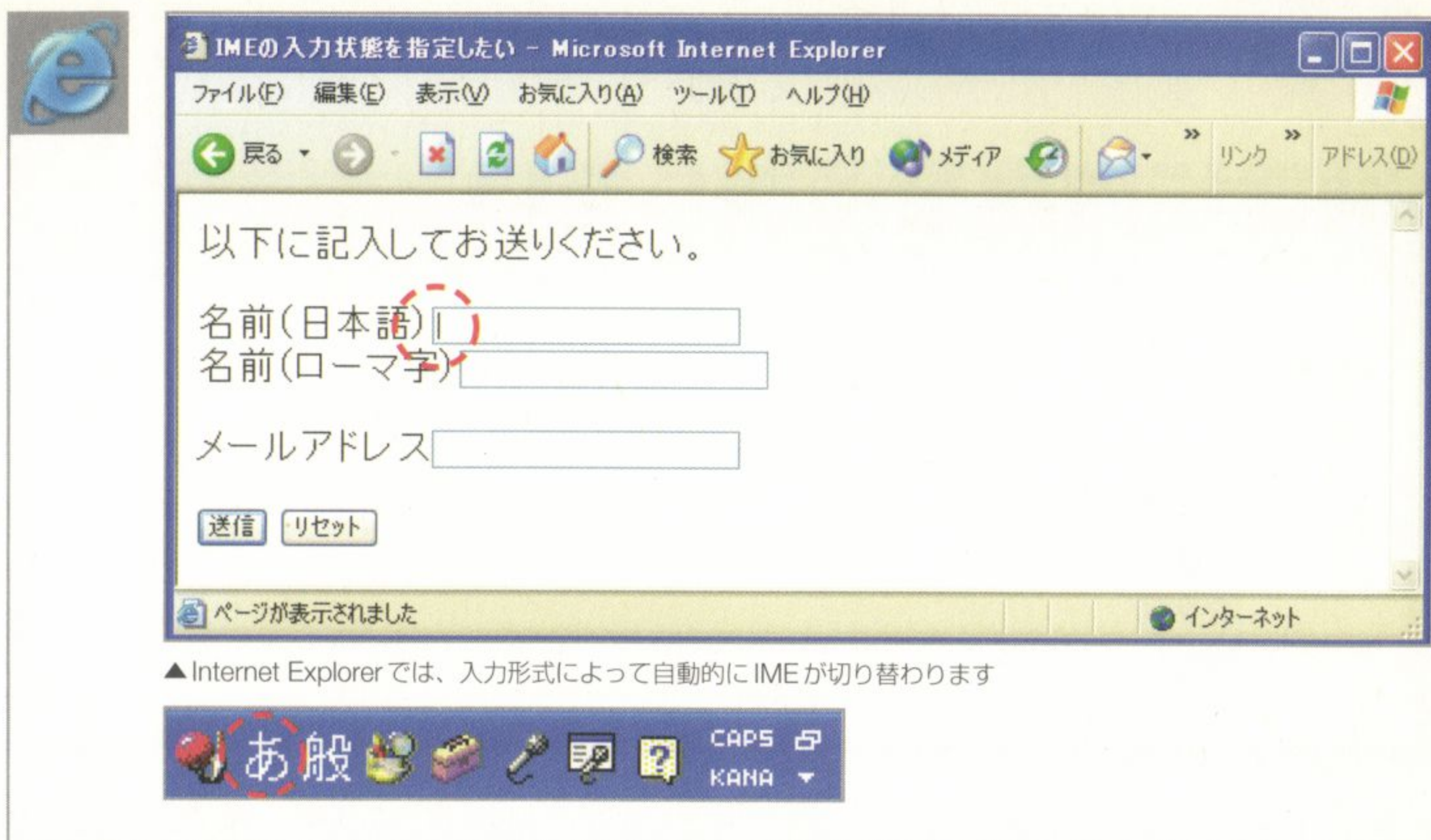
その他

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>IMEの入力状態を指定したい</title>
<style type="text/css">
<!--
input#sample1    { ime-mode: active }
input#sample2    { ime-mode: inactive }
input#sample3    { ime-mode: disabled }
-->
</style>
</head>
<body>
以下に記入してお送りください。
<form>
  <p>
  名前(日本語)<input type="text" size="30" name="n1" id="sample1"><br>
  名前(ローマ字)<input type="text" size="30" name="n2" id="sample2">
  </p>
  <p>
  メールアドレス<input type="text" size="30" name="mail" id="sample3">
  </p>
  <input type="submit" value="送信">
  <input type="reset" value="リセット">
</form>
</body>
</html>
```

その他

▲Internet Explorerでは、入力形式によって自動的にIMEが切り替わります

▲Netscapeは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
auto	×	○	○	○	×	×	×
active	×	○	○	○	×	×	×
inactive	×	○	○	○	×	×	×
disabled	×	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります

※Macintosh版IE5は対応していません

4 OTHER

前後に内容を追加したい

content: ★

★……キーワード
URL
文字列
属性

要素の前後にテキストや画像を挿入するよう設定します。:after擬似要素や、:before擬似要素(p.41参照)とともに使用します。

値には次のような指定方法があります。

キーワード

open-quote	先頭に表示する引用符
close-quote	最後に表示する引用符
no-open-quote	引用符なしの引用の先頭
no-close-quote	引用符なしの引用の最後

open-quoteは引用の開始部、close-quoteは引用の終了部に指定します。デフォルトの引用符はブラウザによりますが(Netscapeの場合は「" "」)、挿入される引用符を変更するにはquotesプロパティを設定します(次項参照)。

no-open-quote、no-close-quoteでは引用符は追加せずに、引用レベルのみ一段深く変更します。no-open-quoteは引用の開始部、no-close-quoteは引用の終了部に指定します。

URL

content: url("☆")　　☆……画像ファイルのURL

追加する画像ファイルのURLを指定します。HTML文書から外部のスタイルシートを読み込む場合はHTML文書からの相対URLではなく、スタイルファイルからの相対URLで指定する必要があります。

文字列

content: "△"　　△……挿入したい文字列

追加する文字列を指定します。

属性

content: attr(◇)　　◇……タグの属性の値

◇に指定したタグの属性の値を表示します。サンプルではalt属性の"顔を出したネコ"という値が画像の後ろに挿入されています。

SOURCE

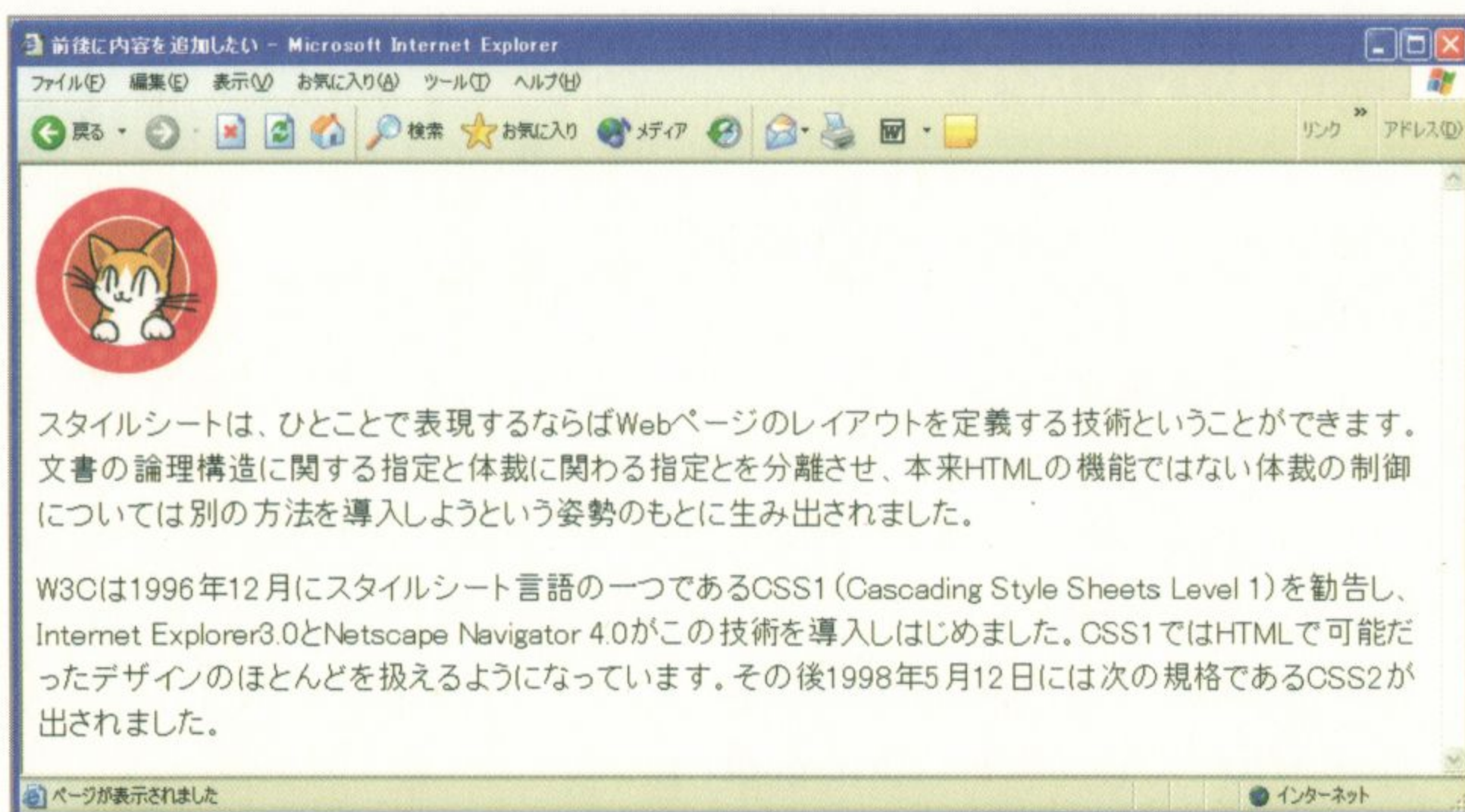
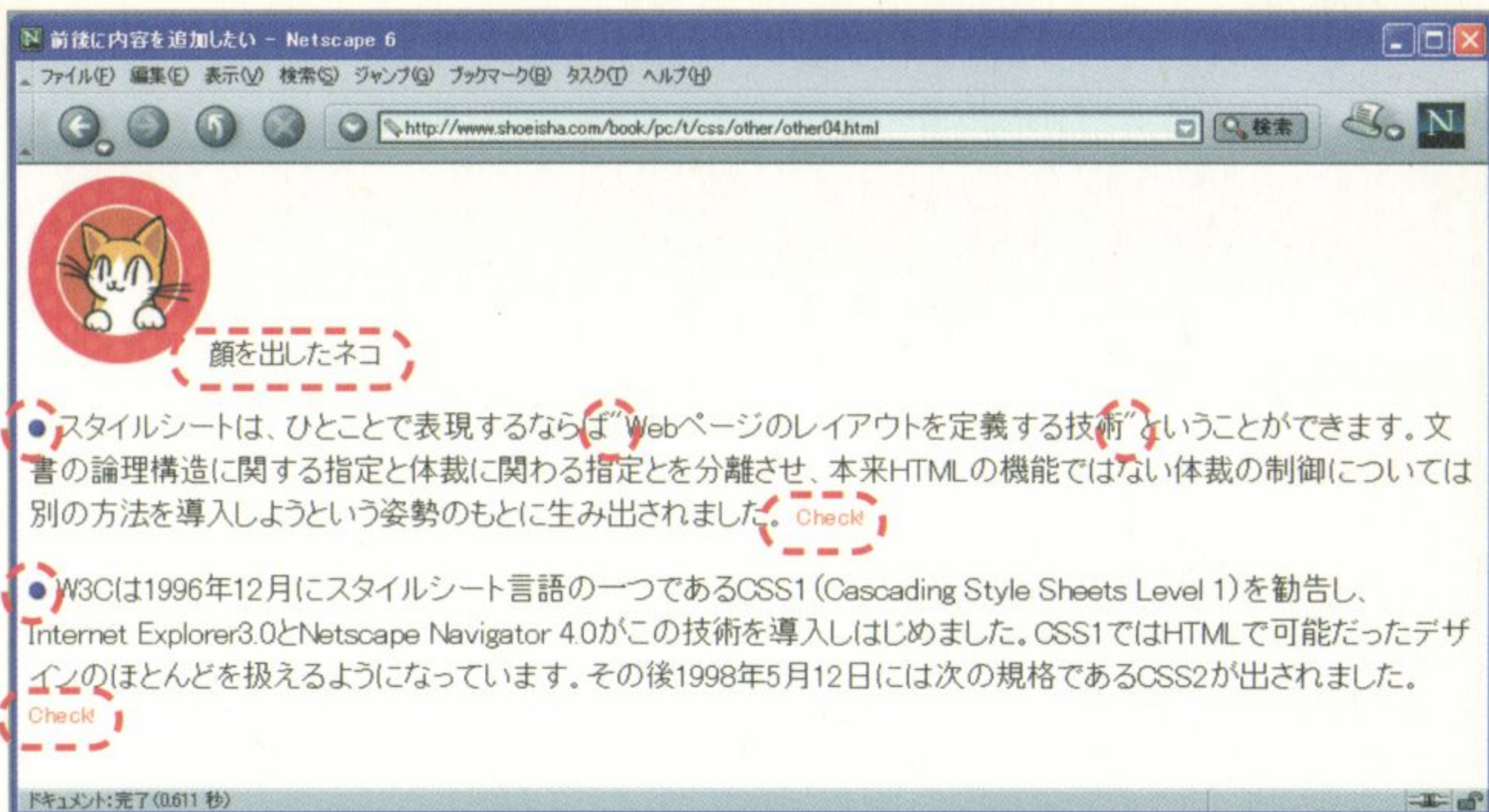
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>前後に内容を追加したい</title>
<style type="text/css">
<!--
p:before      { content: url("ball.gif") }
p:after       {
        content: "Check!";
        font-size: x-small;
        color: red
}
span:before   { content: open-quote }
span:after    { content: close-quote }
img:after     {
        content: attr(alt);
        font-size: small
}
-->
</style>
</head>
<body>
<div><img src="cat1.gif" alt="顔を出したネコ"></div>
<p>
スタイルシートは、ひとことで表現するならば<span>Webページのレイアウトを定義す
る技術</span>ということができます。文書の論理構造に関する指定と体裁に関わる指定
とを分離させ、本来HTMLの機能ではない体裁の制御については別の方法を導入しようとい
```

その他

```
う姿勢のもとに生み出されました。
</p>
<p>
W3Cは1996年12月にスタイルシート言語の一つであるCSS1（Cascading Style Sheets Level 1）を勧告し、Internet Explorer 3.0とNetscape Navigator 4.0がこの技術を導入しはじめました。CSS1ではHTMLで可能だったデザインのほとんどを扱えるようになっています。その後1998年5月12日には次の規格であるCSS2が出されました。
</p>
</body>
</html>
```

▲Internet Explorerは対応していません

その他

その他

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
キーワード	×	×	×	×	×	×	○
URL	×	×	×	×	×	×	○
文字列	×	×	×	×	×	×	○
属性	×	×	×	×	×	×	○

※適用するセレクタによっても効果が変わります

疑似要素 ……………………………………p.40
引用符を指定したい…………………………p.330

引用符を指定したい

quotes: "★" "☆"

quotes: ▲

★……先頭に表示する引用符
☆……最後に表示する引用符
▲……キーワード

contentプロパティ（p.326参照）の値にopen-quoteまたはclose-quoteを指定した場合に、追加される引用符を指定します。

値には次のような指定方法があります。

"先頭の引用符" "後ろの引用符"

先頭に表示する引用符と後ろに表示する引用符を、それぞれ「'」シングルクォーテーションまたは、「"」（ダブルクォーテーション）で囲み、半角スペースで区切って指定します。

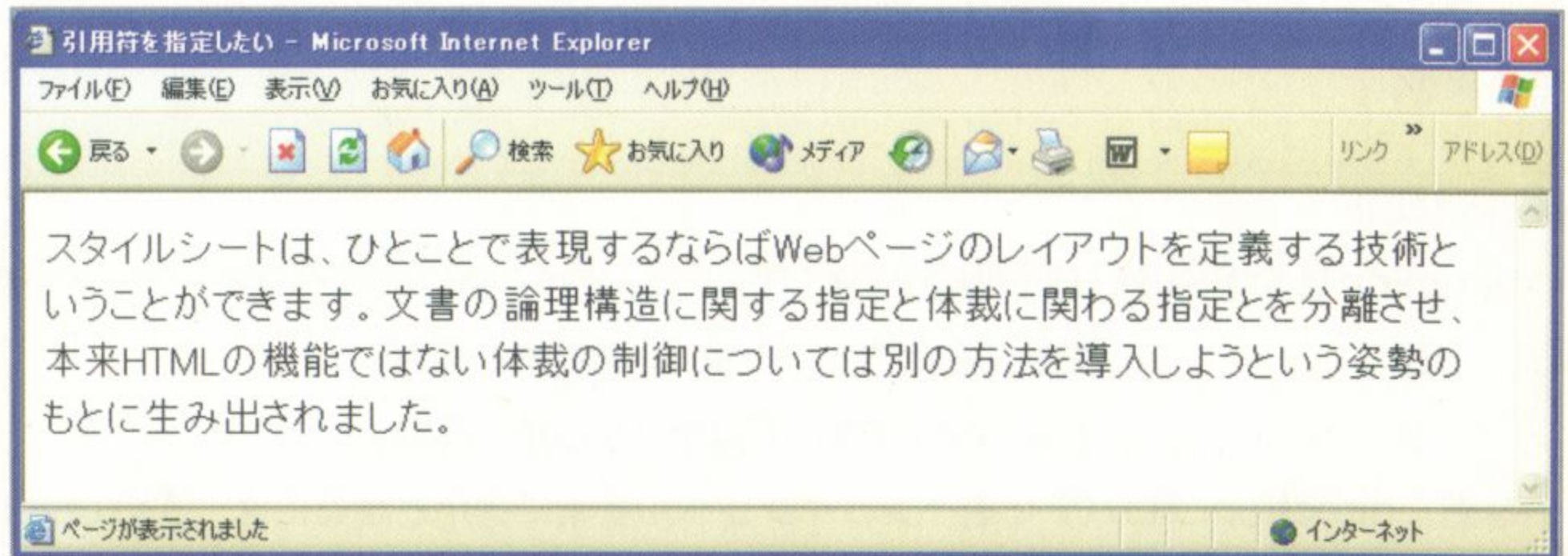
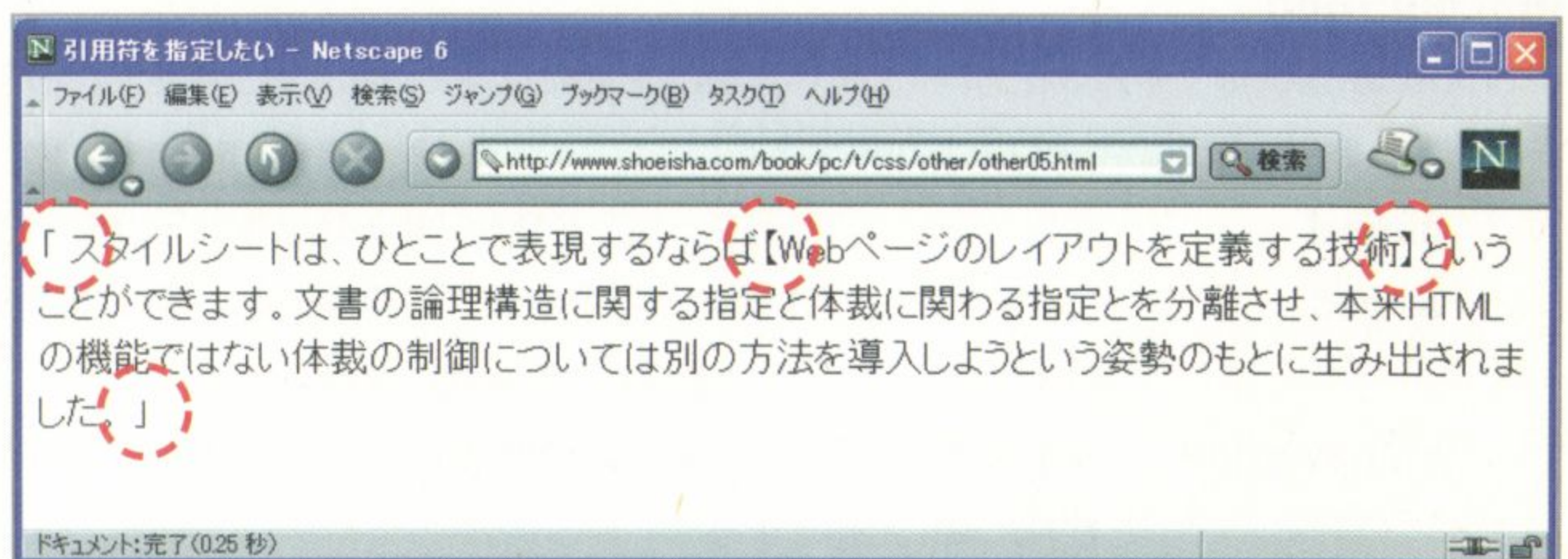
キーワード

none contentプロパティの値にopen-quoteやclose-quoteが指定されていても、引用符を追加しない

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>引用符を指定したい</title>
<style type="text/css">
<!--
body        { line-height: 140% }
p           { quotes: "「" "」" }
p:before    { content: open-quote }
p:after     { content: close-quote }
span        { quotes: "【" "】" }
```

その他

```
span:before    { content: open-quote }
span:after     { content: close-quote }
-->
</style>
</head>
<body>
<p>
スタイルシートは、ひとことで表現するならば<span>Webページのレイアウトを定義する技術</span>ということができます。……（中略）……
</p>
</body>
</html>
```

▲Internet Explorerは対応していません

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
引用符	×	×	×	×	×	×	○

※適用するセレクタによっても効果が変わります

前後に内容を追加したい……………………p.326

6 OTHER

文字の表記方向を指定したい

direction: ★

unicode-bidi: ☆

★……キーワード
☆……キーワード

文字の表記方向を指定し、表記方向の異なる言語を混在できるよう設定します。

たとえば日本語や英語は左から右へと記述しますが、アラビア語やヘブライ語などの文字は右から左へと記述します。directionプロパティとunicode-bidiプロパティで言語の表記方向を指定することにより、こうした表記方向の異なる言語が混在するページを適切に表示できるようになります。

direction

directionは文字の表記方向を指定するプロパティです。値には以下のキーワードがあります。

ltr	左から右の方向（Left To Rightの意味）
rtl	右から左の方向（Right To Leftの意味）

なお、directionプロパティをインラインレベル要素に適用させるには、次のunicode-bodiプロパティの値にembedもしくはbidi-overrideが指定されている必要があります。

unicode-bidi

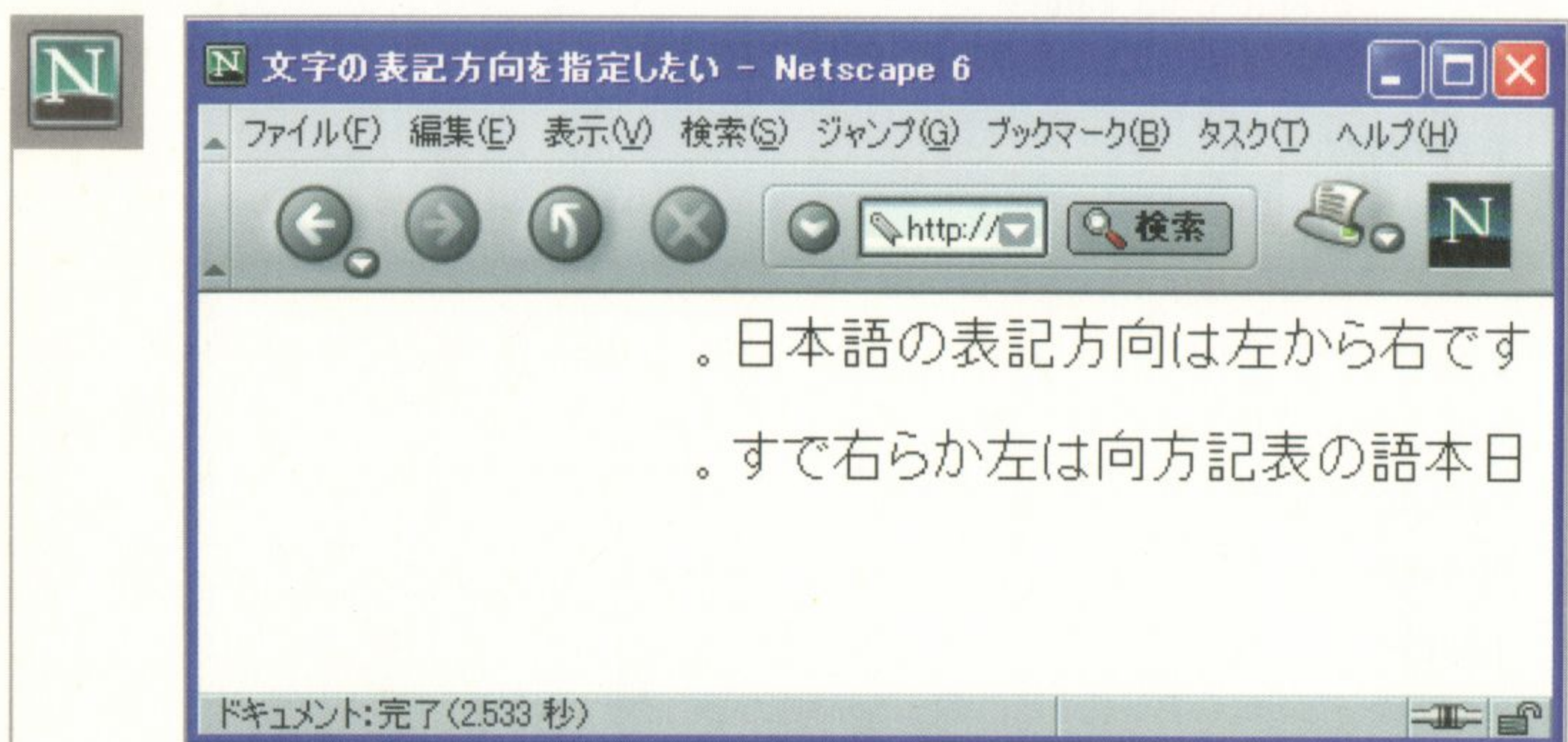
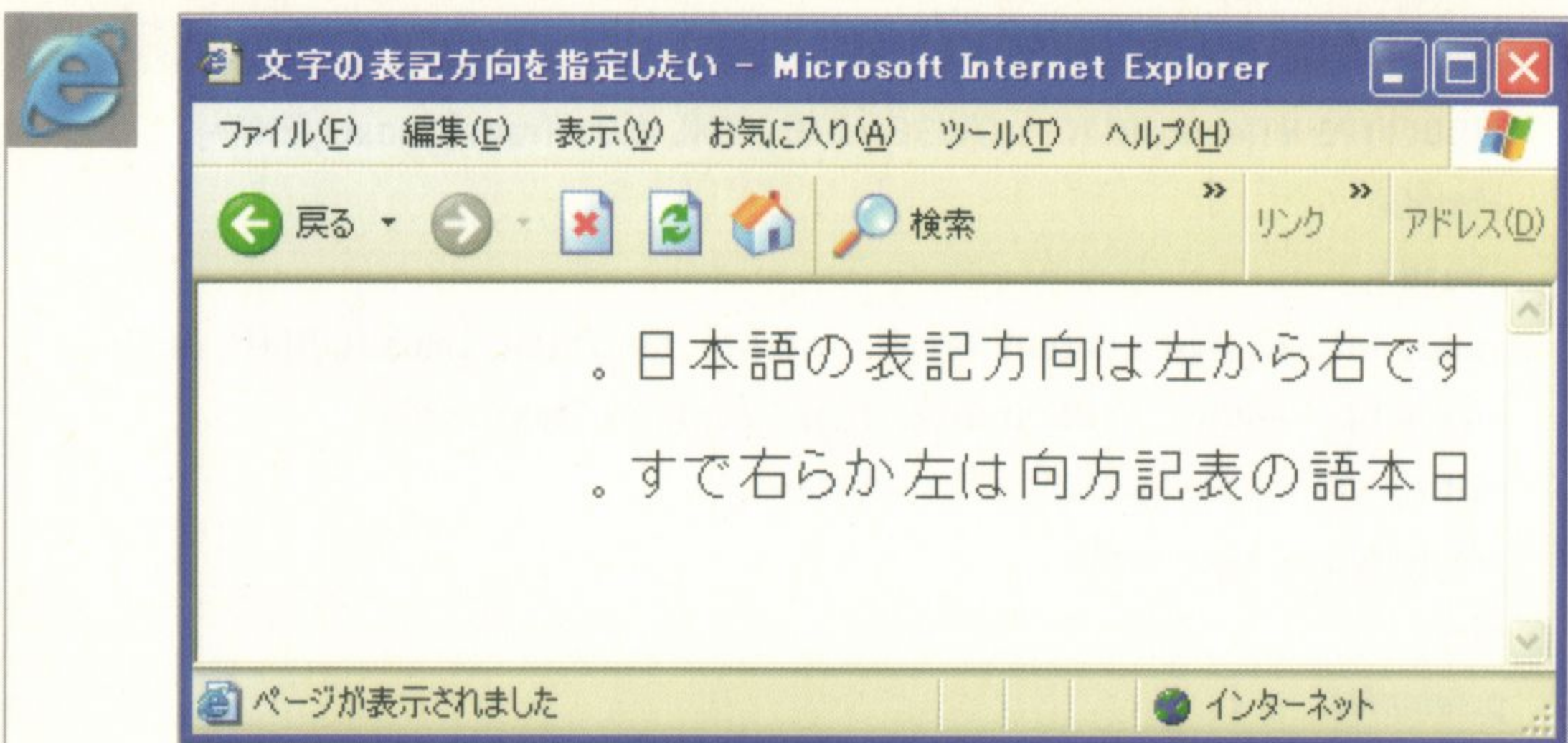
unicode-bidiは文字の表記方向についての指定を新たに埋め込んだり上書きしたりするプロパティです。値には以下のキーワードがあります。

normal	文字の表記方向に関する指定を新たに組み込まず、そのままで表記する
embed	文字の表記方向に関する指定を新たに組み込む。その際の表記方向はdirectionプロパティで設定された値になる
bidi-override	文字の表記方向に関する指定を無効にし、directionプロパティで設定された値に上書きする

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>文字の表記方向を指定したい</title>
<style type="text/css">
<!--
p.sample1      {
        direction: rtl;
        unicode-bidi: embed
}
p.sample2      {
        direction: rtl;
        unicode-bidi: bidi-override
}
-->
</style>
</head>
<body>
<p class="sample1">日本語の表記方向は左から右です。</p>
<p class="sample2">日本語の表記方向は左から右です。</p>
</body>
</html>
```

その他

その他

direction	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
ltr	×	○	○	○	×	×	○
rtl	×	○	○	○	×	×	○

unicode-bidi	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
bidi-override	×	○	○	○	×	×	○
embed	×	○	○	○	×	×	○

印刷時の改ページを指定したい

page-break-after: ★ 直後の改ページ
page-break-before: ★ 直前の改ページ

★……キーワード

印刷時に改ページが行われる位置を指定します。

page-break-beforeは指定した要素の直前の改ページを、page-break-afterは指定した要素の直後の改ページをどのようにするかを指定します。

それぞれ値には以下のキーワードがあります。

always	改ページを行う
auto	改ページを制御しない（デフォルト）
empty-string	改ページを行わない

empty-stringはInternet Explorerが独自に拡張した値です。この値を指定すると、対象となるオブジェクトを印刷する際に要素の前後で改ページをしなくなります。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>印刷時の改ページを指定したい</title>
<style type="text/css">
<!--
h1      { text-align: center }
h2      { page-break-before: always }
hr      { page-break-after: always }
-->
</style>
</head>
<body>
<h1>第一部 スタイルシートの基礎</h1>
<h2>スタイルシートとは</h2>
<p>　Webページの話題のなかで、スタイルシート・CSSという言葉は決してめずらしい
ものではなくなりました。スタイルシートとは、ひとことで表現するならば「Webページ
のレイアウトを定義する技術」ということができるでしょう。</p>
<p>Webページを記述するHTMLは、文書の論理的な構造を示す言語です。……（中略）
……</p>
<p>W3Cは1996年12月に……（中略）……</p>
<hr>
<p>スタイルシートを使いこなすには、HTMLの基礎を理解しておく必要があります。スタ
イルシートについて学習する前に、まずは次の章でHTMLの基本を確認しておきましょう。
</p>
<h2>HTMLの基本</h2>
<p>HTMLの一番基本的な構造を示すと次のようになります。……（中略）……</p>
</body>
</html>
```

第一部 スタイルシートの基礎

スタイルシートとは

Webページの話題のなかで、スタイルシート・CSSという言葉は決してめずらしいものではなくなりました。スタイルシートとは、ひとことで表現するならば「Webページのレイアウトを定義する技術」ということができるでしょう。

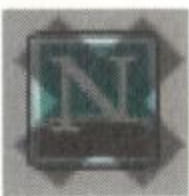
Webページを記述するHTMLは、文書の論理的な構造を示す言語です。

W3Cは1996年12月に

スタイルシートを使いこなすには、HTMLの基礎を理解しておく必要があります。スタイルシートについて学習する前に、まずは次の章でHTMLの基本を確認しておきましょう。

HTMLの基本

HTMLの一番基本的な構造を示すと次のようになります。

▲Internet Explorerの場合、h2要素の直前とhrの直後で自動的に改ページして印刷されます

第一部 スタイルシートの基礎

スタイルシートとは

Webページの話題のなかで、スタイルシート・CSSという言葉は決してめずらしいものではなくなりました。スタイルシートとは、ひとことで表現するならば「Webページのレイアウトを定義する技術」ということができるでしょう。

Webページを記述するHTMLは、文書の論理的な構造を示す言語です。

W3Cは1996年12月に

スタイルシートを使いこなすには、HTMLの基礎を理解しておく必要があります。スタイルシートについて学習する前に、まずは次の章でHTMLの基本を確認しておきましょう。

HTMLの基本

HTMLの一番基本的な構造を示すと次のようになります。

▲Netscapeは対応していないので、改ページの制御は行われません

その他

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
always	×	○	○	○	×	×	×
auto	×	○	○	○	×	×	×
empty-string	×	○	○	○	×	×	×

※適用するセレクタによっても効果が変わります

スクリプトをページから分離させたい

behavior:★

★••••••URL
　　　オブジェクトID

指定した要素に対し、外部スクリプトを適用させます。これにより従来ページに埋め込んでいたスクリプトをページから切り離し、一括管理することも可能になります。

値には次のような指定方法があります。

URL

behavior: url("☆")　☆……外部スクリプトのURL

オブジェクトID

<object>タグのID属性の数字を指定します。<object>タグでなんらかのコントロールを呼び出す場合は、呼び出すコントロールをbehaviorプロパティで扱うことができます。

スクリプトのファイル change.htc

```
<script>
attachEvent("onclick",event_onclick);

function event_onclick(){
        style.color='#ff8800';}
</script>
```

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv="Content-Style-Type" content="text/css">
<title>スクリプトをページから分離させたい</title>
<style type="text/css">
<!--
p       { behavior: url("change.htc") }
}
-->
</style>
</head>
<body>
<p>クリックすると色が変わります。</p>
</body>
</html>
```

	IE4	IE5	IE5.5	IE6	NN4	NN4.7	N6.2
URL	×	○	○	○	×	×	×
オブジェクトID	×	○	○	○	×	×	×

その他

Webページカラーチャート

色の基礎知識

Web配色サンプル

ビジュアルインデックス

適用・デフォルト・継承一覧

スタイルシート乗換一覧

プロパティインデックス

値インデックス

用語インデックス

付録

APPENDIX

1 APPENDIX

Webページカラーチャート

スタイルシートでは、背景や文字などいろいろな要素の色を指定する際に、色を構成する3つの値を使ったRGB値か、色名などのキーワードを使用します。

● 標準16色

この16色はHTML4.01で定義されている色です。これらはWindows VGAのパレットに準拠した色で、色名による指定（p.346）でも正式に使える色となります。

#000080	navy	-	rgb(0,0,128)	rgb(0%,0%,50%)
#0000ff	blue	#00f	rgb(0,0,255)	rgb(0%,0%,100%)
#00ffff	aqua	#0ff	rgb(0,255,255)	rgb(0%,100%,100%)
#008000	green	-	rgb(0,128,0)	rgb(0%,50%,0%)
#00ff00	lime	#0f0	rgb(0,255,0)	rgb(0%,100%,0%)
#ffff00	yellow	#ff0	rgb(255,255,0)	rgb(100%,100%,0%)
#ff0000	red	#f00	rgb(255,0,0)	rgb(100%,0%,0%)
#ff00ff	fuchsia	#f0f	rgb(255,0,255)	rgb(100%,0%,100%)
#800000	maroon	-	rgb(128,0,0)	rgb(50%,0%,0%)
#808000	olive	-	rgb(128,128,0)	rgb(50%,50%,0%)
#008080	teal	-	rgb(0,128,128)	rgb(0%,50%,50%)
#800080	purple	-	rgb(128,0,128)	rgb(50%,0%,50%)
#000000	black	#000	rgb(0,0,0)	rgb(0%,0%,0%)
#808080	gray	-	rgb(128,128,128)	rgb(50%,50%,50%)
#c0c0c0	silver	-	rgb(192,192,192)	rgb(75%,75%,75%)
#ffffff	white	#fff	rgb(255,255,255)	rgb(100%,100%,100%)

● RGB値による指定

コンピュータでは色の3原色である赤、緑、青のそれぞれの強さを0～255までの数値（256段階）で表すことで、特定の色を表現しています。したがってフルカラーと呼ばれるものでは256×256×256=16777216、つまり1677万7216色を扱えるということになります。

#rrggbb方式による指定

ところがコンピュータでは情報はすべて0と1の2進法で表現されるため、数値を2進数で表そうとすると桁数が非常に長くなりがちです。そこで2進数の4桁をまとめ、16進数（0～9の10種類の数字にa～fの6種類のアルファベットを加え、16ごとに桁があがる方式）で数値を表記するようになりました。

色指定値（#rrggbb）方式での記述方法はこの16進数表記にしたがったものです。

「#」につづけて、赤（red）、緑（green）、青（blue）のそれぞれの割合を2桁の16進数（00～ff）で表現し、色を指定します。

たとえば、赤=51、緑=102、青=255の色を16進数で表すと、赤=33、緑=66、青=ffという指定となり、色指定値は「#3366ff」となります。

いくつかの色については実際の#rrggbbの値を掲載していますので参考にしてください。また、10進数と16進数の対応表も掲載しましたので（p.111）、この関係を比較してみるとよいでしょう。

なお、もちろんコンピュータ内部ではすべての数字を2進数に置き換えて処理していることには変わりがありません。

#rgb方式による指定

「#」につづけて、赤（red）、緑（green）、青（blue）のそれぞれの割合を1桁の16進数（0～f）で表現し、計3桁で色を指定します。

この方法ではrgb各桁を2つ繰り返して並べた6桁の形式（#rrggbb）に変換されてから色が表現されます。たとえば「#fb0」という値は「#ffbb00」という値に変換されることになります。

rgb(n,n,n)方式による指定

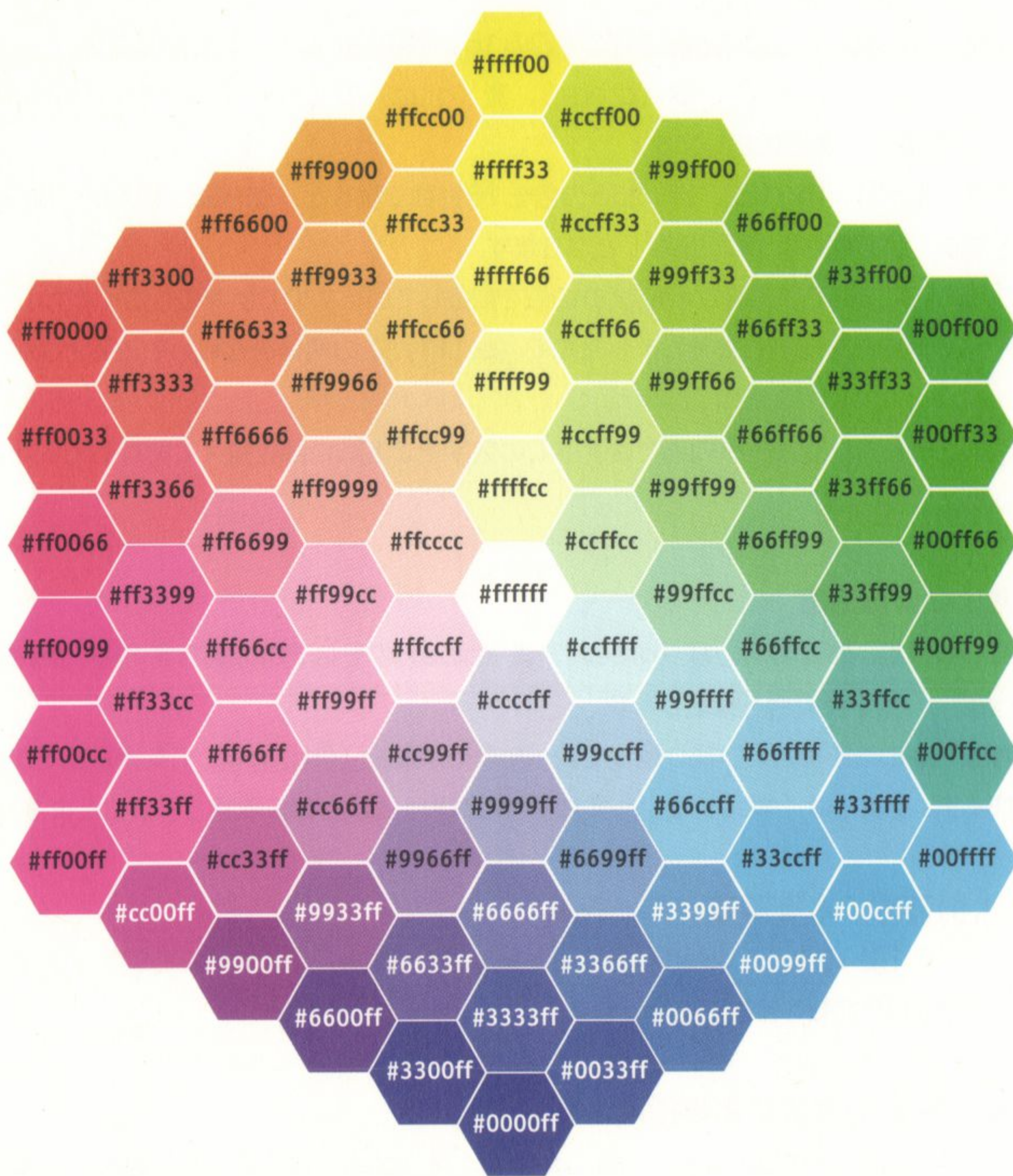
rgbにつづく「()」の中に赤（red）、緑（green）、青（blue）のそれぞれの値を「,」で区切って0から255の10進数の整数で指定します。

rgb(n%,n%,n%)方式による指定

rgbにつづく「()」の中に赤（red）、緑（green）、青（blue）のそれぞれの値を「,」で区切ってパーセントで指定します。

Web Safeカラー

256色の8ビットカラーのうちWindowsとMacintoshのシステムパレットに共通した216色で、Internet ExplorerとNetscapeでもほぼ同じように表示される色をWeb Safeカラーといいます。これらの色は16進数で表現した場合、RGBの各値が 00、33、66、99、cc、ff の組み合わせからできています。Web Safeカラーを使えば、たとえユーザーの環境が256色表示であっても比較的問題なく正しく表示されますが、これ以外の色を使った場合は自動的にディザリング処理（近い色に置き換えられる）されることになり、意図したとおりの色で見てもらえなくなる可能性もあるので注意が必要です。

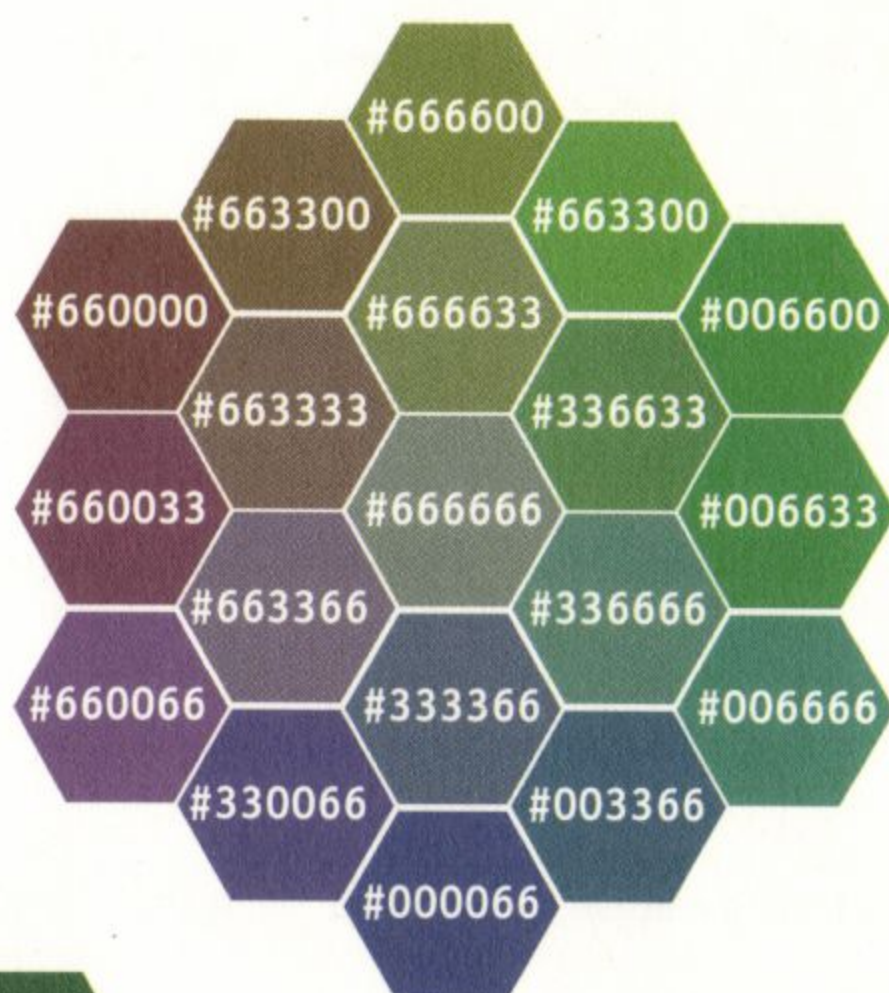
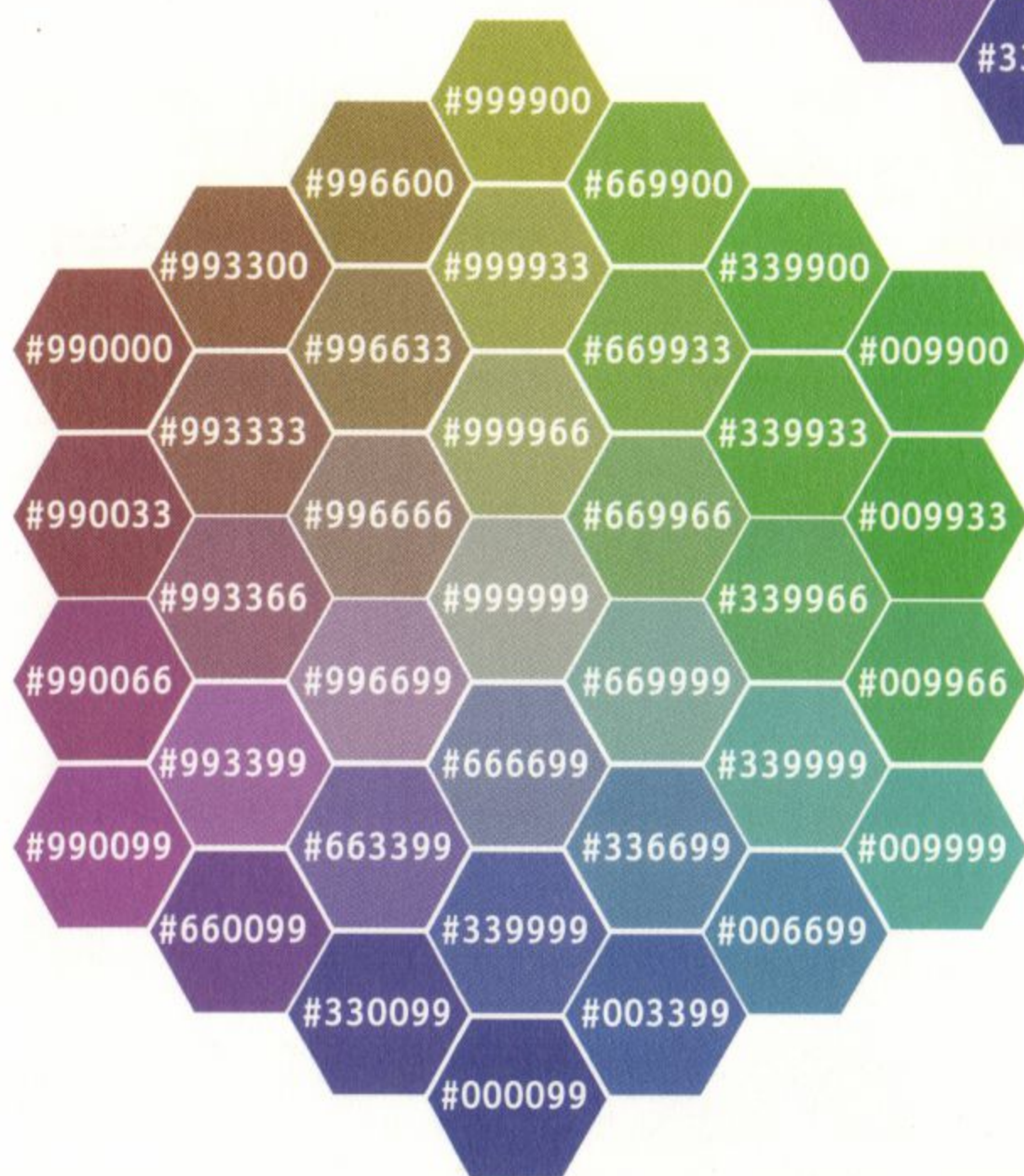
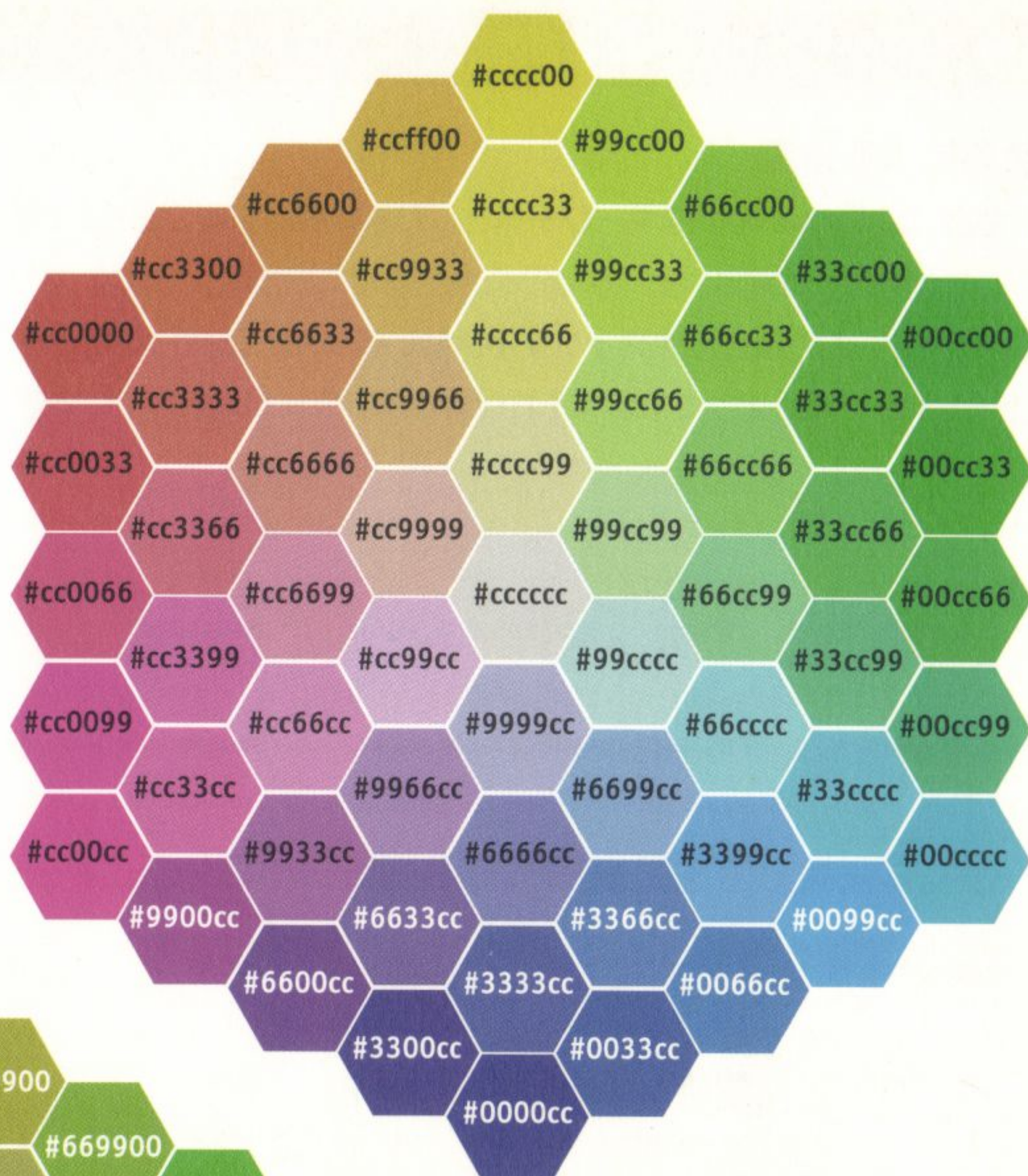
下記の図はWeb Safeカラーを#ffffffから#000000の明度の異なるグレーを基準にして色相環状に配列したものです。

▲ #ffffffを基準としたWeb Safeカラーの色相環

▲#ccccccを基準としたWeb Safeカラーの色相環

▲#999999を基準としたWeb Safeカラーの色相環

▲#666666を基準とした
Web Safeカラーの色相環

▲#333333を基準とした
Web Safeカラーの色相環

▲#000000

● キーワードによる指定

色名による指定

色名（colorname）をキーワードとして色を指定することもできます。

この場合は { color: red } のように色名を値に直接記入します（「"」で囲まないように注意してください）。この場合、大文字小文字は問いません。正式に使用できる色は★印のついている16色（p.342参照）だけですが、それ以外でブラウザが対応している色もあります。

色名	16進	RGB
blanchedalmond	#ffebcd	rgb(255,235,205)
lightyellow	#ffffe0	rgb(255,255,224)
cornsilk	#fff8dc	rgb(255,248,220)
antiquewhite	#faebd7	rgb(250,235,215)
papayawhip	#ffefd5	rgb(255,239,213)
lemonchiffon	#fffacd	rgb(255,250,205)
beige	#f5f5dc	rgb(245,245,220)
linen	#faf0e6	rgb(250,240,230)
oldlace	#fdf5e6	rgb(253,245,230)
lightcyan	#e0ffff	rgb(224,255,255)
aliceblue	#f0f8ff	rgb(240,248,255)
whitesmoke	#f5f5f5	rgb(245,245,245)
lavenderblush	#fff0f5	rgb(255,240,245)
floralwhite	#fffaf0	rgb(255,250,240)
mintcream	#f5fffa	rgb(245,255,250)
ghostwhite	#f8f8ff	rgb(248,248,255)
honeydew	#f0fff0	rgb(240,255,240)
seashell	#fff5ee	rgb(255,245,238)
ivory	#fffff0	rgb(255,255,240)
azure	#f0ffff	rgb(240,255,255)
snow	#fffafa	rgb(255,250,250)
★ white	#ffffff	rgb(255,255,255)
gainsboro	#dcdcdc	rgb(220,220,220)
lightgrey	#d3d3d3	rgb(211,211,211)
★ silver	#c0c0c0	rgb(192,192,192)
darkgray	#a9a9a9	rgb(169,169,169)
lightslategray	#778899	rgb(119,136,153)
slategray	#708090	rgb(112,128,144)
★ gray	#808080	rgb(128,128,128)
dimgray	#696969	rgb(105,105,105)
darkslategray	#2f4f4f	rgb(47,79,79)
★ black	#000000	rgb(0,0,0)

lawngreen	#7cfc00	rgb(124,252,0)		mediumspringgreen	#00fa9a	rgb(0,250,154)
greenyellow	#adff2f	rgb(173,255,47)	★	teal	#008080	rgb(0,128,128)
chartreuse	#7fff00	rgb(127,255,0)		darkcyan	#008b8b	rgb(0,139,139)
★ lime	#00ff00	rgb(0,255,0)		lightseagreen	#20b2aa	rgb(32,178,170)
limegreen	#32cd32	rgb(50,205,50)		mediumaquamarine	#66cdaa	rgb(102,205,170)
yellowgreen	#9acd32	rgb(154,205,50)		cadetblue	#5f9ea0	rgb(95,158,160)
★ olive	#808000	rgb(128,128,0)		steelblue	#4682b4	rgb(70,130,180)
olivedrab	#6b8e23	rgb(107,142,35)		aquamarine	#7fffd4	rgb(127,255,212)
darkolivegreen	#556b2f	rgb(85,107,47)		powderblue	#b0e0e6	rgb(176,224,230)
forestgreen	#228b22	rgb(34,139,34)		paleturquoise	#afeeee	rgb(175,238,238)
darkgreen	#006400	rgb(0,100,0)		lightblue	#add8e6	rgb(173,216,230)
★ green	#008000	rgb(0,128,0)		lightsteelblue	#b0c4de	rgb(176,196,222)
seagreen	#2e8b57	rgb(46,139,87)		skyblue	#87ceeb	rgb(135,206,235)
mediumseagreen	#3cb371	rgb(60,179,113)		lightskyblue	#87cefa	rgb(135,206,250)
darkseagreen	#8fbc8b	rgb(143,188,143)		mediumturquoise	#48d1cc	rgb(72,209,204)
lightgreen	#90ee90	rgb(144,238,144)		turquoise	#40e0d0	rgb(64,224,208)
palegreen	#98fb98	rgb(152,251,152)		darkturquoise	#00ced1	rgb(0,205,209)
springgreen	#00ff7f	rgb(0,255,127)	★	aqua	#00ffff	rgb(0,255,255)

Name	Hex	RGB
★ red	#ff0000	rgb(255,0,0)
crimson	#dc143c	rgb(220,20,60)
★ maroon	#800000	rgb(128,0,0)
darkred	#8b0000	rgb(139,0,0)
brown	#a52a2a	rgb(165,42,42)
sienna	#a0522d	rgb(160,82,45)
saddlebrown	#8b4513	rgb(139,69,19)
indianred	#cd5c5c	rgb(205,92,92)
rosybrown	#bc8f8f	rgb(188,143,143)
lightcoral	#f08080	rgb(240,128,128)
salmon	#fa8072	rgb(250,128,114)
darksalmon	#e9967a	rgb(233,150,122)
coral	#ff7f50	rgb(255,127,80)
tomato	#ff6347	rgb(255,99,71)
sandybrown	#f4a460	rgb(244,164,96)
lightsalmon	#ffa07a	rgb(255,160,122)
lightsalmon	#ffa07a	rgb(255,160,122)
chocolate	#d2691e	rgb(210,105,30)

Name	Hex	RGB
orangered	#ff4500	rgb(255,69,0)
orange	#ffa500	rgb(255,165,0)
darkorange	#ff8c00	rgb(255,140,0)
tan	#d2b48c	rgb(210,180,140)
peachpuff	#ffdab9	rgb(255,218,185)
bisque	#ffe4c4	rgb(255,228,196)
moccasin	#ffe4b5	rgb(255,228,181)
navajowhite	#ffdead	rgb(255,222,173)
wheat	#f5deb3	rgb(245,222,179)
burlywood	#deb887	rgb(222,184,135)
darkgoldenrod	#b8860b	rgb(184,134,011)
goldenrod	#daa520	rgb(218,165,032)
gold	#ffd700	rgb(255,215,0)
★ yellow	#ffff00	rgb(255,255,0)
lightgoldenrodyellow	#fafad2	rgb(250,250,210)
palegoldenrod	#eee8aa	rgb(238,232,170)
khaki	#f0e68c	rgb(240,230,140)
darkkhaki	#bdb76b	rgb(189,183,107)

cyan	#00ffff	rgb(0,255,255)
deepskyblue	#00bfff	rgb(0,191,255)
dodgerblue	#1e90ff	rgb(30,144,255)
cornflowerblue	#6090ef	rgb(96,144,239)
royalblue	#4169e1	rgb(65,105,225)
★ blue	#0000ff	rgb(0,0,255)
mediumblue	#0000cd	rgb(0,0,205)
★ navy	#000080	rgb(0,0,128)
darkblue	#00008b	rgb(0,0,139)
midnightblue	#191970	rgb(25,25,112)
darkslateblue	#483d8b	rgb(72,61,139)
slateblue	#6a5acd	rgb(106,90,205)
mediumsateblue	#0e00b0	rgb(14,0,176)
mediumpurple	#9370db	rgb(147,112,219)
darkorchid	#9932cc	rgb(153,50,204)
darkviolet	#9400d3	rgb(148,0,211)
blueviolet	#8a2be2	rgb(138,43,226)
mediumorchid	#ba55d3	rgb(186,85,211)
plum	#dda0dd	rgb(221,160,221)
lavender	#e6e6fa	rgb(230,230,250)
thistle	#d8bfd8	rgb(216,191,216)
orchid	#da70d6	rgb(218,112,214)
violet	#ee82ee	rgb(238,130,238)
indigo	#4b0082	rgb(75,0,130)
darkmagenta	#8b008b	rgb(139,0,139)
★ purple	#800080	rgb(128,0,128)
mediumvioletred	#c71585	rgb(199,21,133)
deeppink	#ff1493	rgb(255,20,147)
★ fuchsia	#ff00ff	rgb(255,0,255)
magenta	#ff00ff	rgb(255,0,255)
hotpink	#ff69b4	rgb(255,105,180)
palevioletred	#db7093	rgb(219,112,147)
lightpink	#ffb6c1	rgb(255,182,193)
pink	#ffc0cb	rgb(255,192,203)
mistyrose	#ffe4e1	rgb(255,228,225)

システムカラーによる指定

スタイルシートでは、WindowsやMacOSが保持しているシステム情報を呼び出すことができます。このシステムカラーを使うと、ページを見る人のOSの環境に合わせて使用色を決めることができます（p.48参照）。

システムカラーには以下のキーワードがあります。

activeborder	アクティブなウィンドウの枠の色
activecaption	アクティブなウィンドウのタイトルバーの色
appworkspace	アプリケーション内のウィンドウの背景色
background	デスクトップの背景色
buttonface	立体的ボタンの表面の色
buttonhighlight	立体的なボタンの光のあたっている面の色
buttonshadow	立体的なボタンの影になってる面の色
buttontext	立体的なボタンのテキストの色
captiontext	タイトルバーのテキストの色
graytext	選択できないテキストの色
highlight	選択している状態の色
highlighttext	選択しているテキストの色
inactiveborder	アクティブでないウィンドウの枠の色
inactivecaption	アクティブでないウィンドウのタイトルバーの色
inactivecaptiontext	アクティブでないウィンドウのタイトルバーのテキストの色
infobackground	ツールチップの背景色
infotext	ツールチップのテキストの色
menu	メニューの背景色
menutext	メニューのテキストの色
scrollbar	スクロールバーの色
threeddarkshadow	立体的に表示される部分の暗い影の色
threedface	立体的に表示される部分の表面の色
threedhighlight	立体的に表示される部分の光のあたっている面の色
threedlightshadow	立体的に表示される部分の明るい影の色
threedshadow	立体的に表示される部分の影の色
window	ウィンドウの背景色
windowframe	ウィンドウのフレームの色
windowtext	ウィンドウのテキストの色

transparentの指定

プロパティによってはtransparent（透明）を指定できるものもあります。transparentを指定するとその領域は透明になり、結果としてその要素が含まれる親ボックスの（つまり下の）内容や背景・背景画像などが透けて見えるようになります。

16進数対応表

10進数と16進数の対応表です。色を表す256段階の数値（左側）は、16進数で表すと右側の数値となります。赤字の部分はWeb Safeカラーを構成する数値です。

10進数	16進数	10進数	16進数	10進数	16進数	10進数	16進数	10進数	16進数	10進数	16進数	10進数	16進数	10進数	16進数
0	00	32	20	64	40	96	60	128	80	160	a0	192	c0	224	e0
1	01	33	21	65	41	97	61	129	81	161	a1	193	c1	225	e1
2	02	34	22	66	42	98	62	130	82	162	a2	194	c2	226	e2
3	03	35	23	67	43	99	63	131	83	163	a3	195	c3	227	e3
4	04	36	24	68	44	100	64	132	84	164	a4	196	c4	228	e4
5	05	37	25	69	45	101	65	133	85	165	a5	197	c5	229	e5
6	06	38	26	70	46	102	66	134	86	166	a6	198	c6	230	e6
7	07	39	27	71	47	103	67	135	87	167	a7	199	c7	231	e7
8	08	40	28	72	48	104	68	136	88	168	a8	200	c8	232	e8
9	09	41	29	73	49	105	69	137	89	169	a9	201	c9	233	e9
10	0a	42	2a	74	4a	106	6a	138	8a	170	aa	202	ca	234	ea
11	0b	43	2b	75	4b	107	6b	139	8b	171	ab	203	cb	235	eb
12	0c	44	2c	76	4c	108	6c	140	8c	172	ac	204	cc	236	ec
13	0d	45	2d	77	4d	109	6d	141	8d	173	ad	205	cd	237	ed
14	0e	46	2e	78	4e	110	6e	142	8e	174	ae	206	ce	238	ee
15	0f	47	2f	79	4f	111	6f	143	8f	175	af	207	cf	239	ef
16	10	48	30	80	50	112	70	144	90	176	b0	208	d0	240	f0
17	11	49	31	81	51	113	71	145	91	177	b1	209	d1	241	f1
18	12	50	32	82	52	114	72	146	92	178	b2	210	d2	242	f2
19	13	51	33	83	53	115	73	147	93	179	b3	211	d3	243	f3
20	14	52	34	84	54	116	74	148	94	180	b4	212	d4	244	f4
21	15	53	35	85	55	117	75	149	95	181	b5	213	d5	245	f5
22	16	54	36	86	56	118	76	150	96	182	b6	214	d6	246	f6
23	17	55	37	87	57	119	77	151	97	183	b7	215	d7	247	f7
24	18	56	38	88	58	120	78	152	98	184	b8	216	d8	248	f8
25	19	57	39	89	59	121	79	153	99	185	b9	217	d9	249	f9
26	1a	58	3a	90	5a	122	7a	154	9a	186	ba	218	da	250	fa
27	1b	59	3b	91	5b	123	7b	155	9b	187	bb	219	db	251	fb
28	1c	60	3c	92	5c	124	7c	156	9c	188	bc	220	dc	252	fc
29	1d	61	3d	93	5d	125	7d	157	9d	189	bd	221	dd	253	fd
30	1e	62	3e	94	5e	126	7e	158	9e	190	be	222	de	254	fe
31	1f	63	3f	95	5f	127	7f	159	9f	191	bf	223	df	255	ff

モニタと印刷の色の違い

原理的に、印刷ではWeb上（モニタで見る色）の色を完全に再現することはできません。ここに記載された色はあくまでも参考にとどめ、実際にお使いになる場合は、モニタ上で色を確認してください。また、細かい色のニュアンスは、Webページを見に来る人のモニタの環境によって大きく異なる場合もあるので、注意が必要です。

なお、本書のWebページでは、ここに掲載しているカラーチャートを実際にWeb上でご覧いただくことができます。

http://www.shoeisha.com/book/pc/dic/

2 APPENDIX

色の基礎知識

ここでは、Webの配色を考える際に参考になる、色に関する基本的な知識を紹介します。

● 色の属性

色の属性について理解しておくと、配色を考えやすくなります。色には「色相」「明度」「彩度」の3つの属性があります。

色相

白～灰色～黒を無彩色といい、それ以外の色を有彩色と言います。「色相」とは、それぞれの有彩色が持つ色合いのことです。似た色相を隣合わせに並べていくと、色の輪ができます。これは「色相環」と呼ばれ、このうちおおよそ赤～黄の範囲の色を「暖色」、緑～青の範囲の色を「寒色」と表現しています。

色相環上で、相対する位置にある色を「補色」といいます。補色関係にある2色を並べると、強い対比が生じ、緊張感のあるはっきりした配色になります。また、色相環上で約120度の位置にある色を「準補色」といいます。補色による配色ではどぎついという場合には、準補色同士を並べると、ゆるやかな対比を作ることができます。

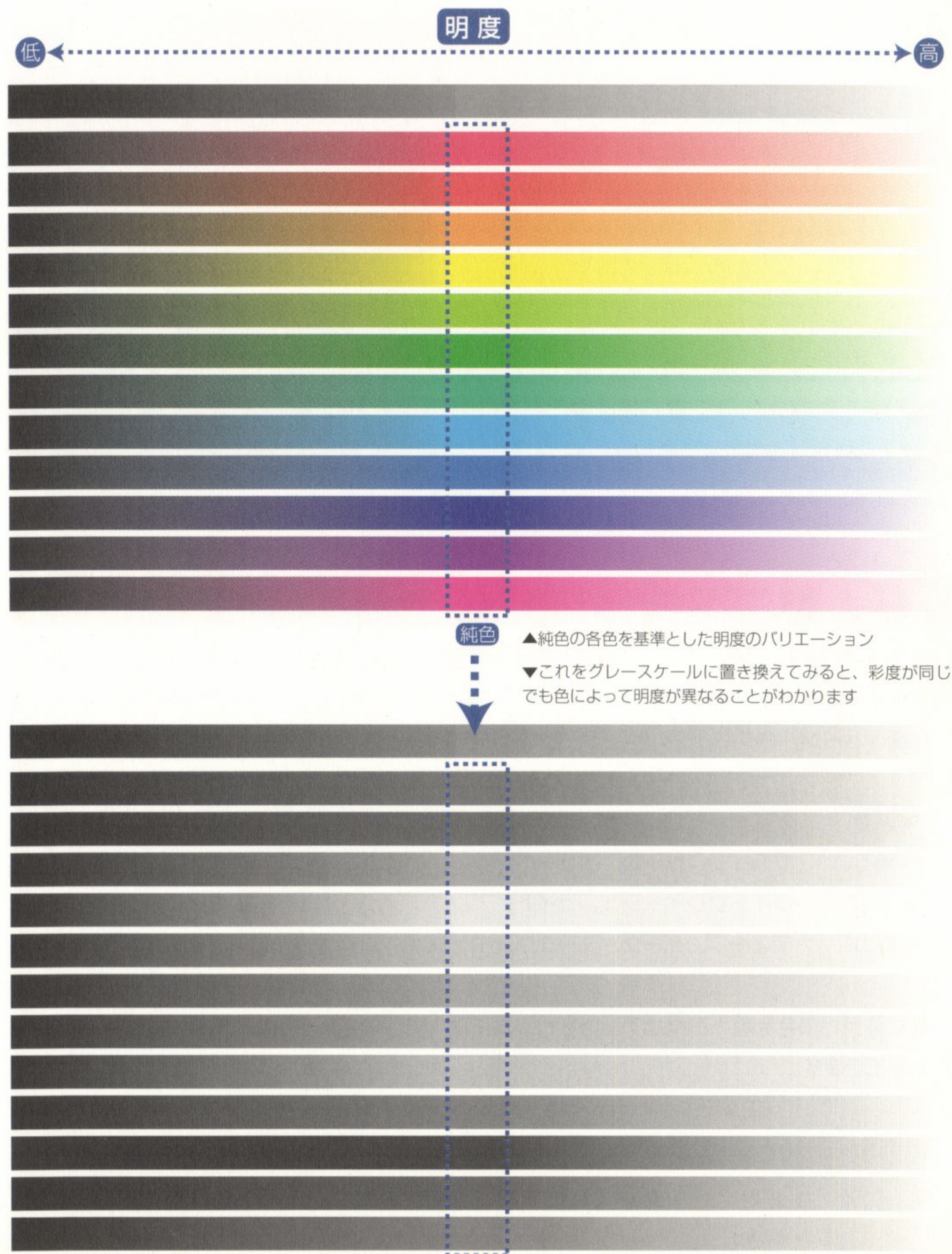
▲色相環上で相対する位置にある2色を、「補色関係にある色」といいます

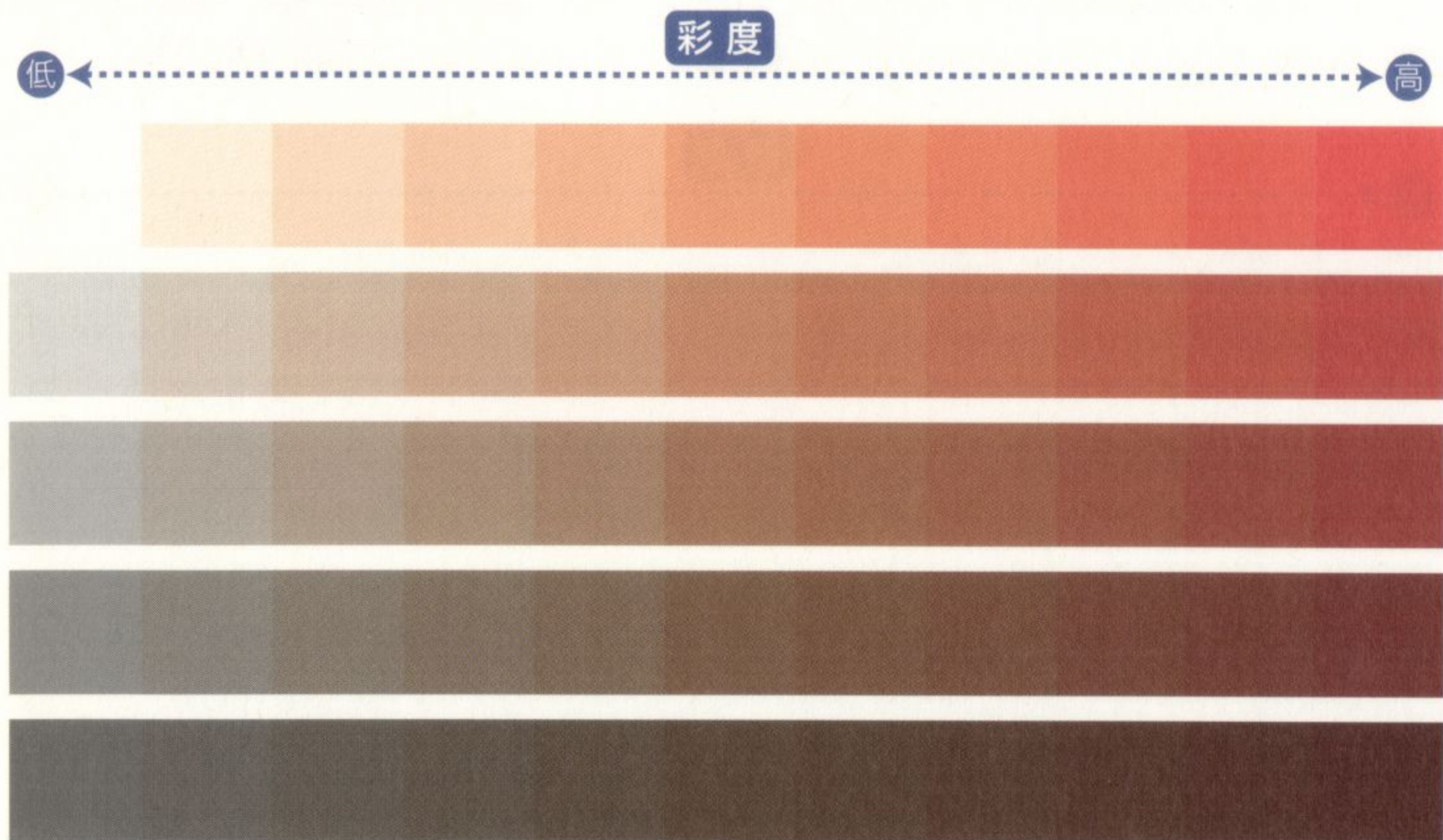
明度

「明度」は色の明るさの度合いのことで、白から黒までのグレースケールを基準としています。白に近づくほど明度は高く、黒に近づくほど明度は低くなります。たとえば、赤に白を混ぜたピンクは、元の赤より明るい色（＝高明度）です。一方、赤に黒を混ぜた茶色は、元の赤より暗い色（＝低明度）になります。

また、純色（彩度が最高の色）の赤と黄をグレースケールに置き換えてみると、黄より赤の方が暗い灰色になります。このように、同じ彩度であっても、色相によって明度は異なります。

▲純色の各色を基準とした明度のバリエーション

▼これをグレースケールに置き換えてみると、彩度が同じでも色によって明度が異なることがわかります

彩度

「彩度」は色の鮮やかさ（色みの強さ）の度合いのことで、無彩色の彩度は0になります。純色の赤に白や黒などの無彩色を混ぜていくと、だんだん色みが薄れて無彩色に近づき、彩度は低くなっていきます。混色された無彩色の分量が少なくて純色（彩度が最高の色）に近いほど色みが強く、彩度は高くなります。

▲無彩色の混色が少ないほど彩度が高くなります。この図では、右上がもっとも彩度の高い色です

● 色調（トーン）

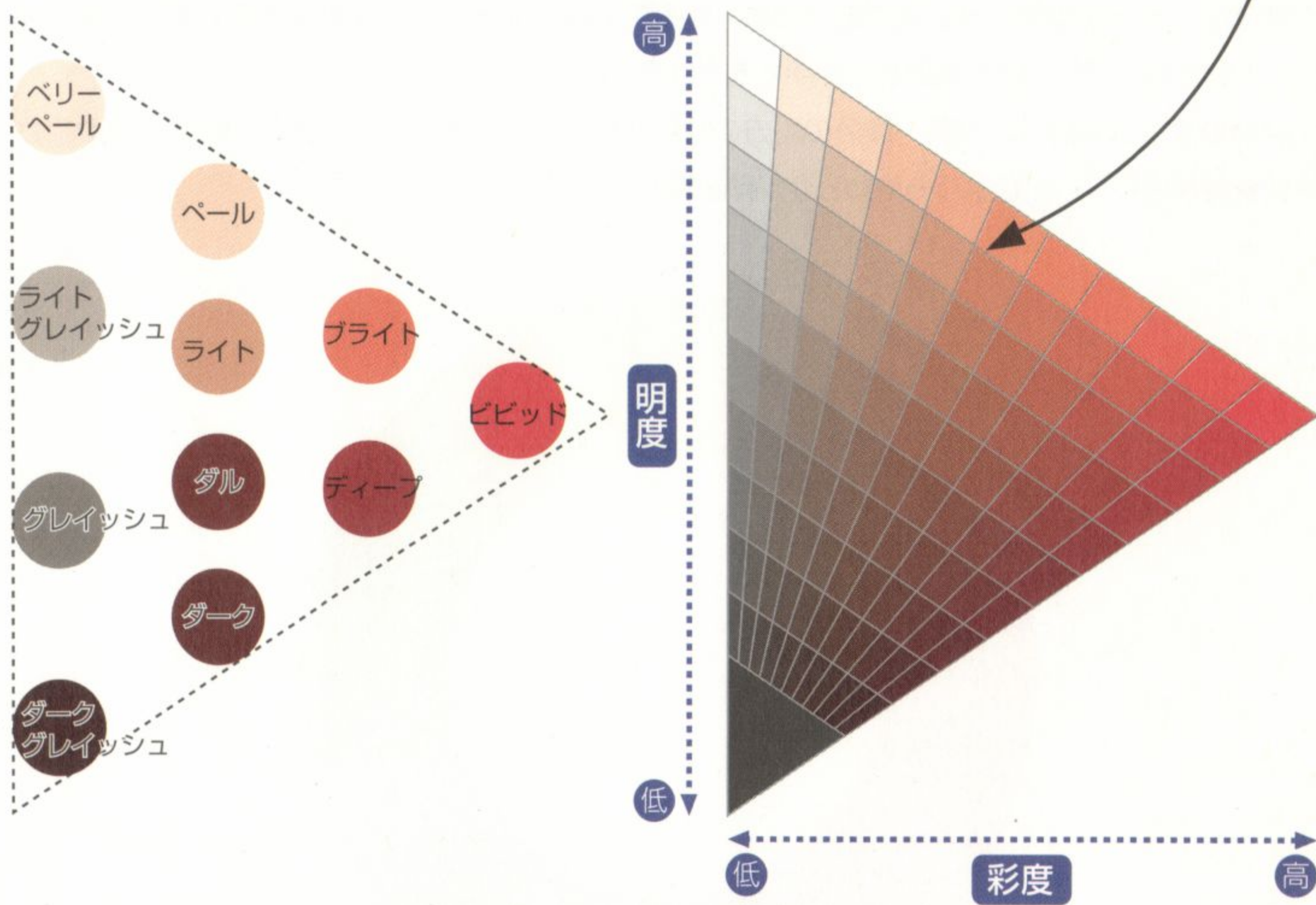
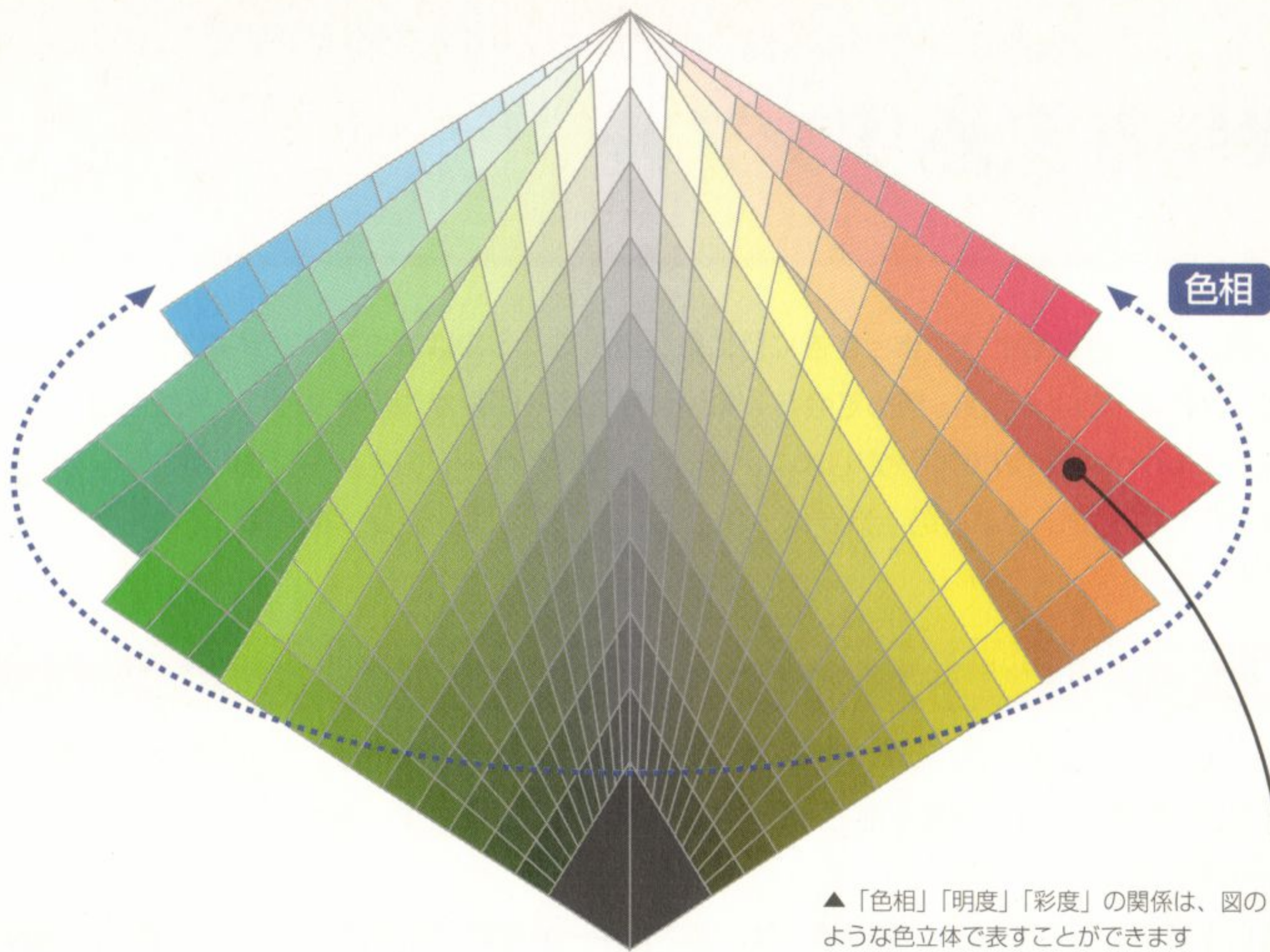
色の3つの属性を総合して、色の分布を示した図を「色立体」といいます。色立体の中から、ある色相に関する部分を取り出し、明度・彩度に応じて分類すると、1つの色相内の色は次のような色調（トーン）のグループに分けることができます。

- 派手　ビビッド
- 明るい　ブライト、ペール、ベリーペール
- 地味　ライトグレイッシュ、ライト、グレイッシュ、ダル
- 暗い　ディープ、ダーク、ダークグレイッシュ

異なる色相の色を組み合わせて配色を行う際には、各色の色調を揃えておくと上手くまとめることができます。

▲「色相」「明度」「彩度」の関係は、図のような色立体で表すことができます

▲明度と彩度を組み合わせた「色の調子」を「色調（トーン）」といいます

3 APPENDIX

Web配色サンプル

Webページをみるとき、まっさきに目に飛び込んでくるのは、コンテンツよりもまずページの「色」ではないでしょうか。初めて会う人の服装から第一印象が決まるように、私たちは、まず色をみてWebページの印象を決定します。作りたいWebページのイメージを明確にし、効果的な配色を行うことで、サイトの主旨がはっきりし、より深くコンテンツを理解してもらうことが可能になります。

● 赤～黄系の配色

赤～黄系は、「暖色」と呼ばれる色の系統です。一般に暖色系の色は、外向的で生命や情熱、親しさなどを象徴します。熱量を感じさせ食欲をそそる色であるため、飲食関連のWebページには欠かせません。

特に赤色は「炎」や「血」を連想させ、エネルギーや生命力に溢れた色です。闘争心・勇気・興奮などを伝える一方、熱狂や怒りなどの不安定なニュアンスや、強い禁止を表すためにも利用されます。

黄色は「光」を連想させる色であるため、陽気で健康なイメージ、幸福感を表します。また金色に近いため、華やかさや高貴さ、派手さを表す色でもあります。

中間のオレンジ色は、赤色・黄色の両方の性質を持っています。強い主張の中に親しみやすさや爽快感が加わり、赤色よりもやわらかい印象になります。

同系色の配色

#ff6699	#ff0099	#ff0033
#ff66ff	#ff0000	#ff9900
#ff9966	#ff6600	#ffcc00
#ffff66	#ffff00	#99ff00

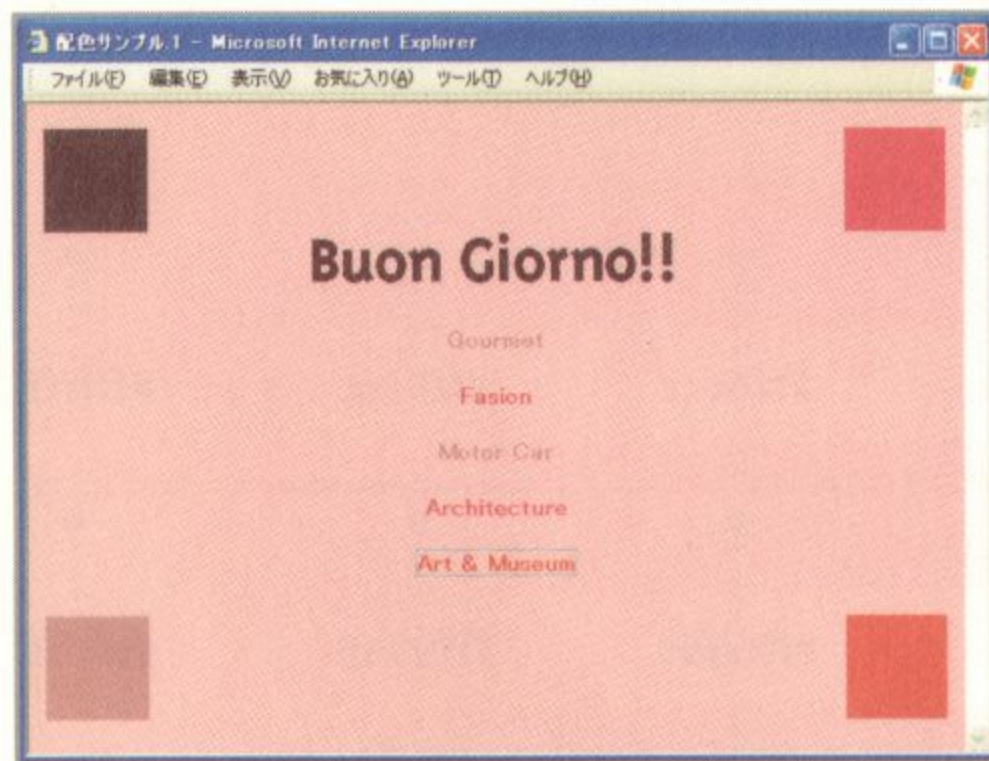
#cc6699	#cc0066	#cc0033
#cc6666	#cc0000	#cc3300
#cc9966	#cc6600	#ccff00
#cccc66	#cccc00	#99cc00

補色・準補色との対比

#ff0099	#00ff66
#ff0000	#00ffff
#ff6600	#0099ff
#ffff00	#0000ff

#ff0099	#33ff00
#ff0000	#00ff66
#ff6600	#00ffcc
#ffff00	#0099ff

#ff0099	#00ffff
#ff0000	#00ff66
#ff6600	#00ffff
#ffff00	#9900ff

明度のバリエーション

▲同系色による配色は、全体をまとめやすく、落ち着いた印象になります

▲補色を加えると、ポイントが強調され、躍動感が生まれます

▲高明度・高彩度の同系色でまとめると、明るく穏やかな印象になります

▲明度差や補色を利用すると、強い主張が感じられるようになります

● 緑～青系の配色

緑～青系は、「寒色」と呼ばれる色の系統です。暖色に比べて内向的で、理知や抑制を象徴します。

緑色は、「植物」の色。草木を見ると心がなごむように、緑色には穏やかで落ち着いた雰囲気を作る効果があります。また、新緑の季節のような清涼感や、新鮮な野菜、自然界のバランスなども連想させます。中庸で安定した印象のため色自体の自己主張は少なく、ポイントカラーを引き立てるベースカラーとして機能することが多くなっています。

自然界のどこででも目にするようでいて、実体を持つ青いものは少ないことから、青色には抽象的でさまざまなイメージが託されます。まず、「空」や「海」の色であることから、爽快感、広がりや永遠、穏やかさ、神秘性などが連想されます。フレッシュでスポーティな色であり、ノーブル、フォーマルを象徴し、憂鬱や悲壮感を表すこともあります。このようにイメージに幅はありますが、青色は理性や冷静さが基本となっています。

同系色の配色

#ccff66	#99ff00	#33ff00
#66ff66	#00ff00	#00ff66
#66ffcc	#00ff99	#00ffcc
#66ffff	#00ffff	#0099ff

#99cc66	#66cc00	#33cc00
#66cc66	#00cc00	#00cc33
#66cc99	#00cc66	#00cc99
#66cccc	#00cccc	#0099cc

補色・準補色との対比

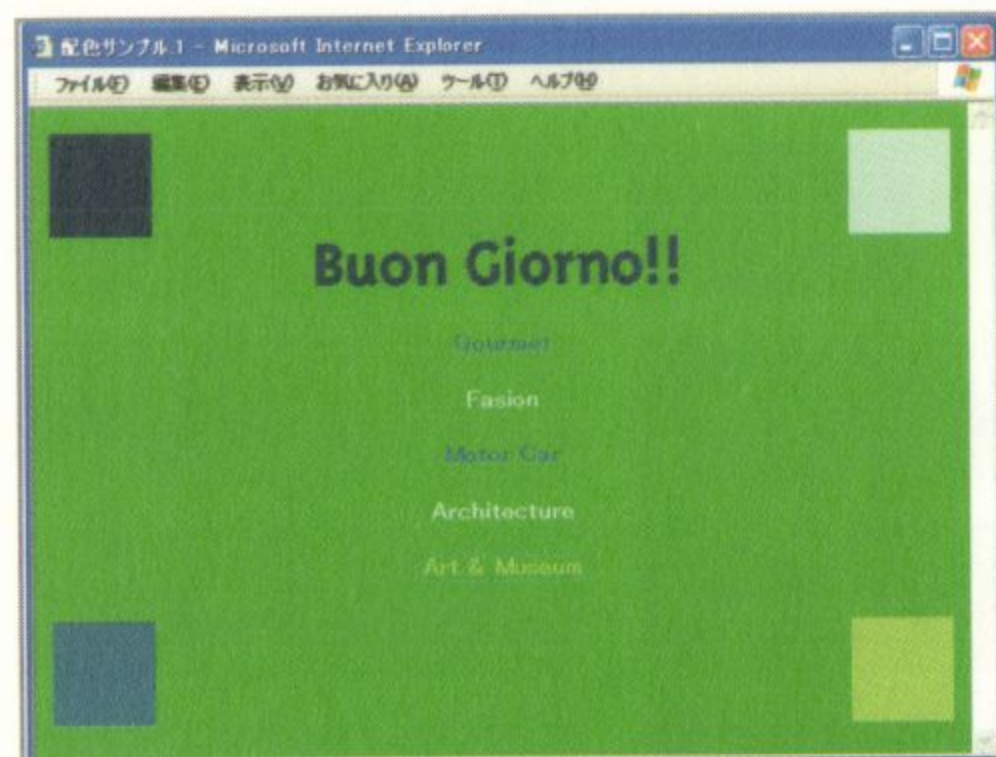
#99ff00	#0033ff
#00ff00	#6600ff
#00ff99	#ff00ff
#00ffff	#ff0099

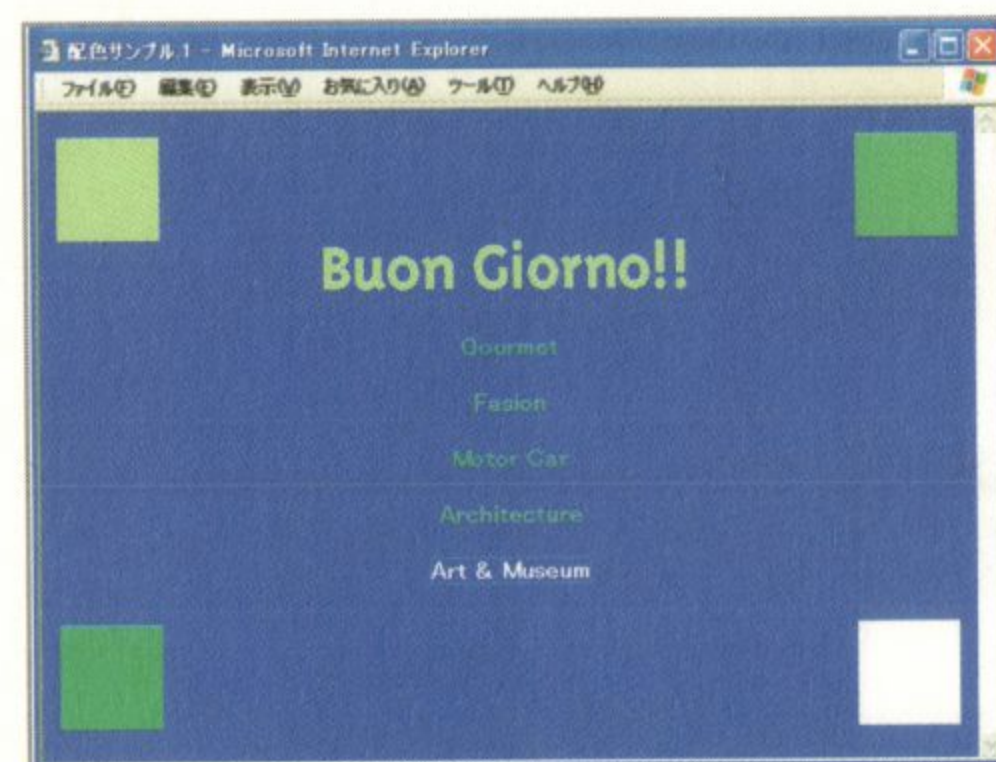
#99ff00	#ff00ff
#00ff00	#ff0066
#00ff99	#ff3300
#00ffff	#ff9900

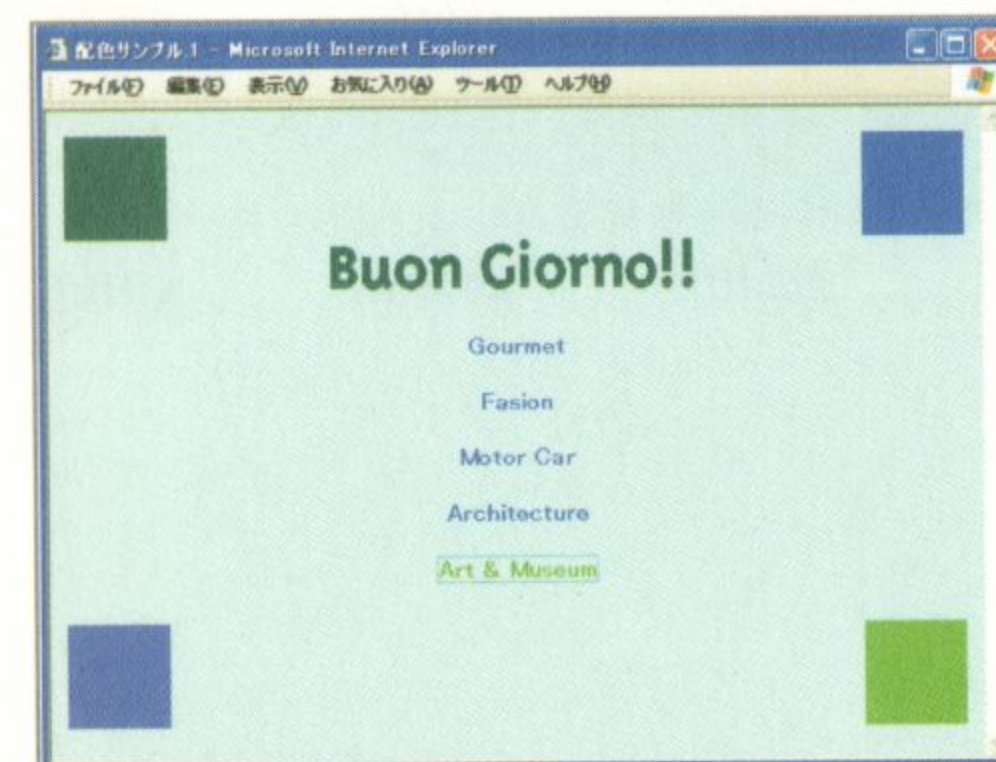
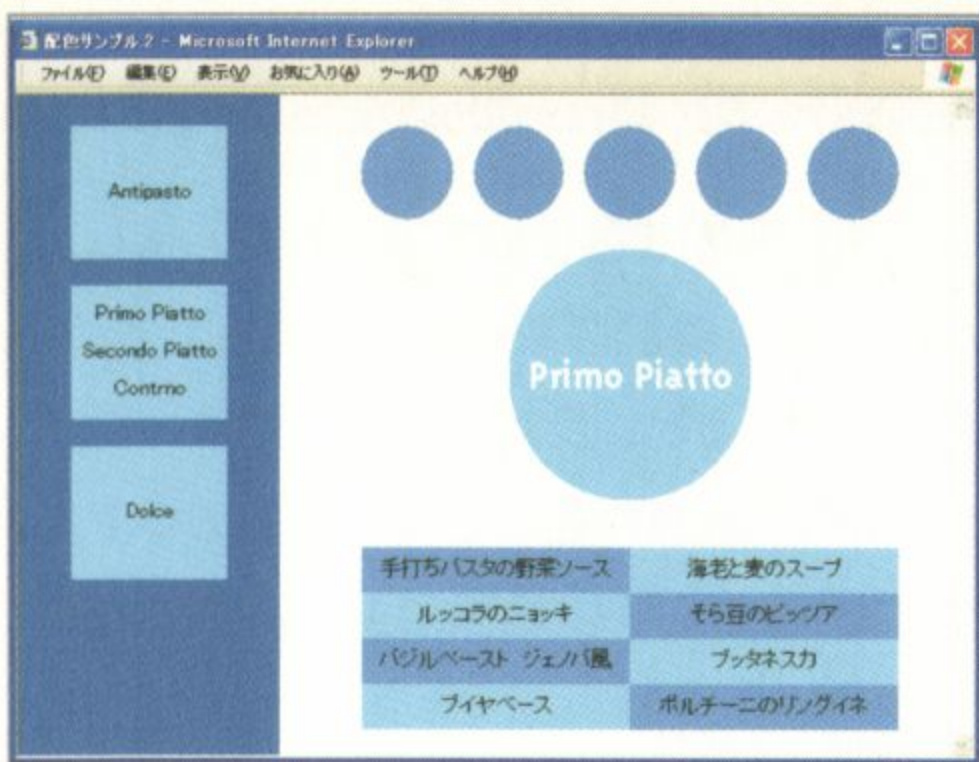
明度のバリエーション

配色サンプル1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

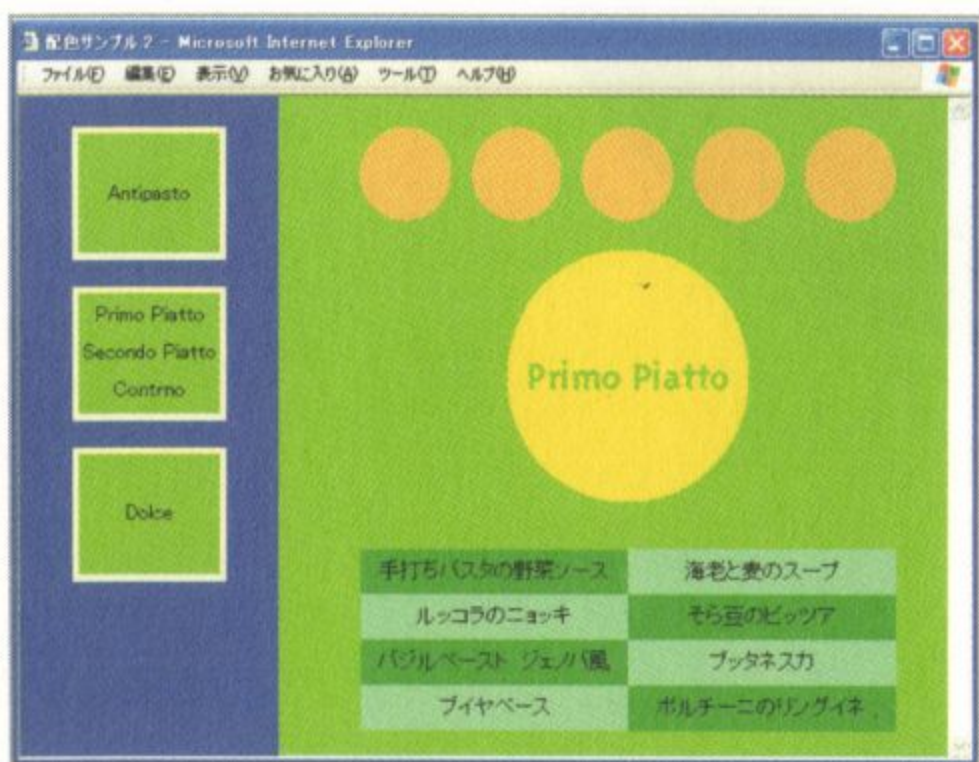
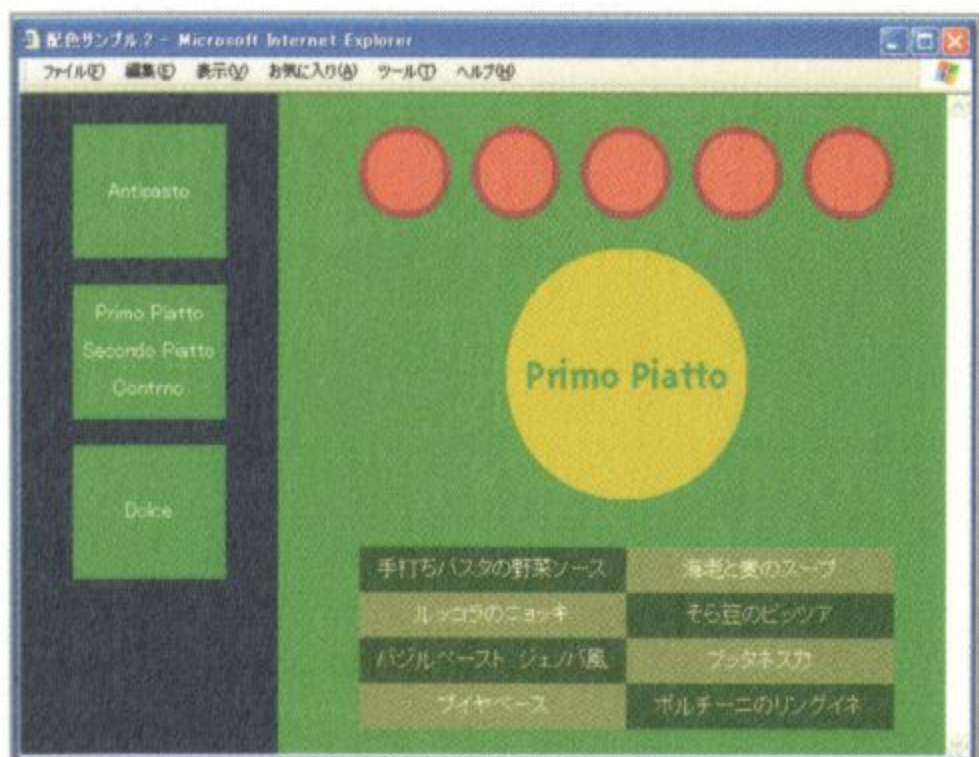
#ccff99 bgcolor
#336666 text
#999900 link
#339900 vlink
#339999 alink

配色サンプル1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Fasion
Architecture

#33cc33 bgcolor
#003366 text
#ccffcc link
#006699 vlink
#ccff66 alink

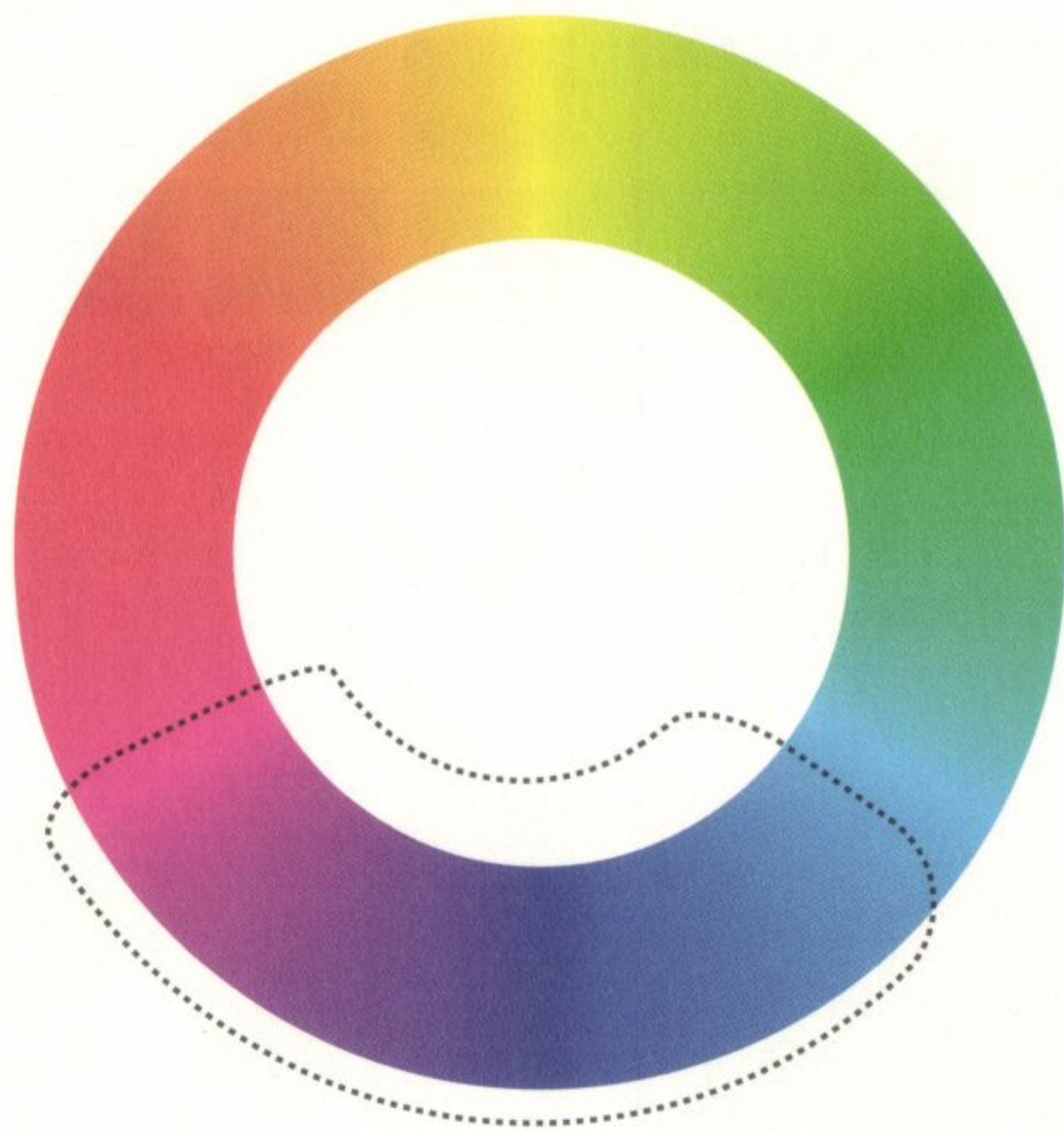
配色サンプル1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Art & Museum

#3333ff bgcolor
#ccff99 text
#00ff99 link
#00cc99 vlink
#ffffff alink

配色サンプル1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#ccffff bgcolor
#006666 text
#0066ff link
#6666ff vlink
#99cc66 alink

#0066cc #99ccff #ffffff #6699ff

▲同系色による配色は、全体をまとめやすく、落ち着いた印象になります

▲全色相による配色を取り入れると、明るく賑やかな印象になります

▲彩度の高い清色を組み合わせると、若々しくスピード感が生まれます

▲明度・彩度が低くなるほど、枯れて地味な印象になります

● 紫～赤紫系の配色

自然界に少ない紫色は、古来より神秘性や非日常性、高貴さを表す色として扱われてきました。権力を象徴し、退廃や爛熟、病的、狂気を表す色でもあります。高級感や気品、優雅さ、華麗など、大人っぽく色気のあるイメージを持ちますが、多用しすぎると、反対に下品、陰気、派手、くどい、怪しいなどのマイナスイメージを作ることになるので注意が必要です。紫色の中でも、赤みの深いワイン色などは豊かな実りを連想させますが、明るく派手な紫色は食品関連では好まれません。

同系色の配色

#6699ff	#0066ff	#0033ff
#6666ff	#0000ff	#6600ff
#9966ff	#9900ff	#cc00ff
#ff66ff	#ff00ff	#ff00cc

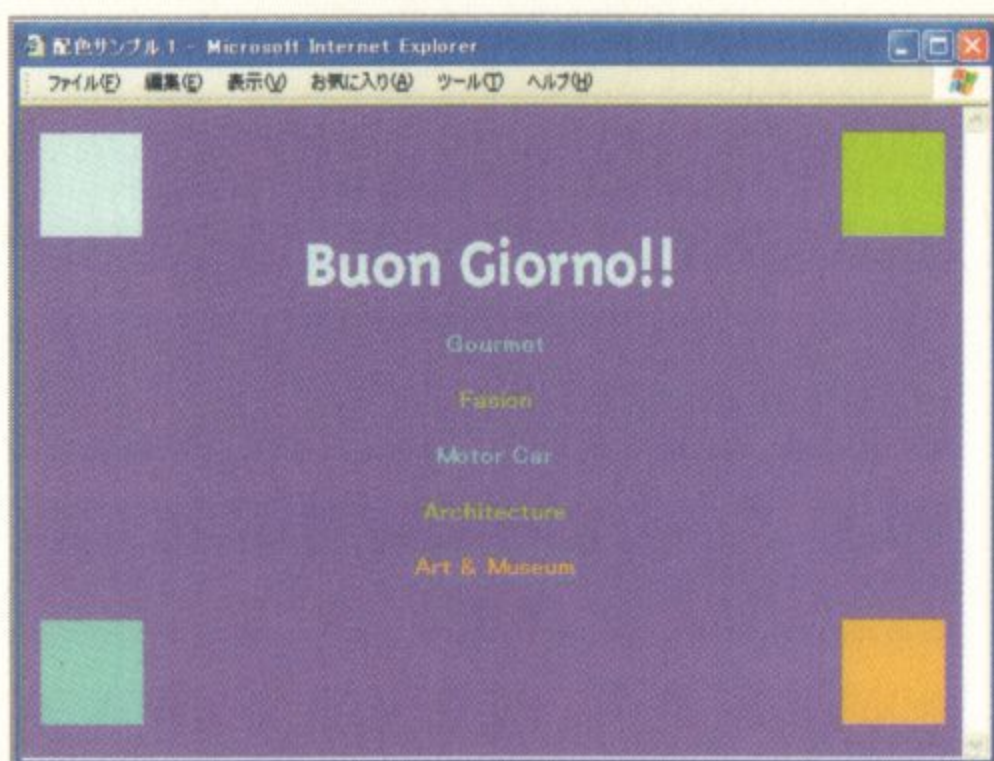
#3366cc	#0066cc	#0033cc
#3333cc	#0000cc	#3300cc
#9933cc	#6600cc	#9900cc
#cc33cc	#cc00cc	#cc0099

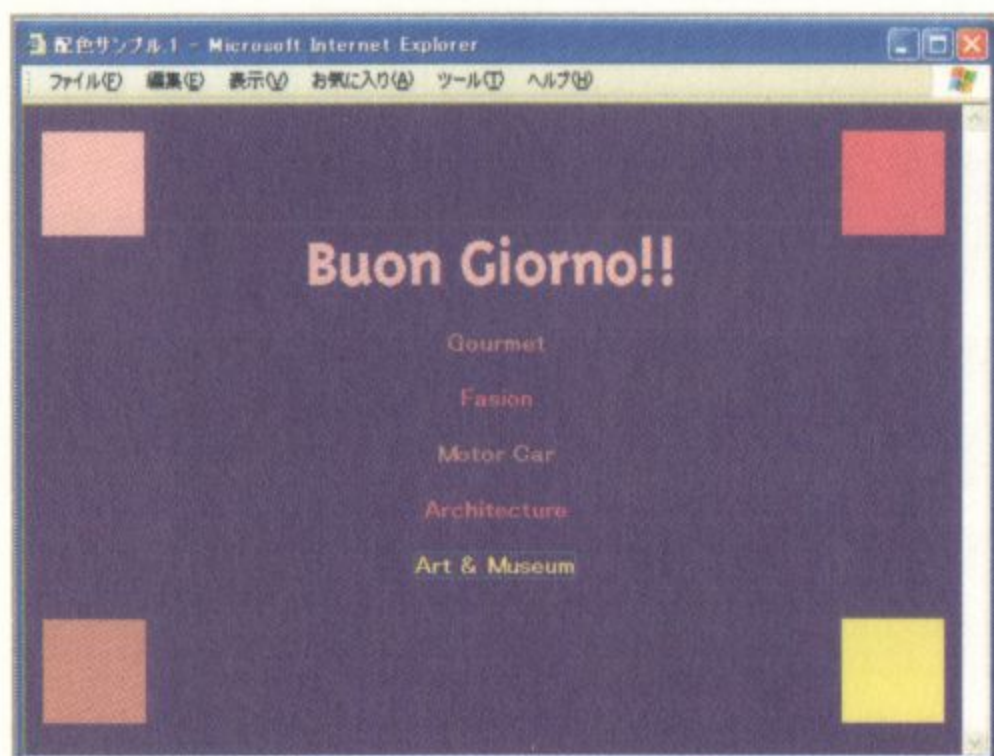
補色・準補色との対比

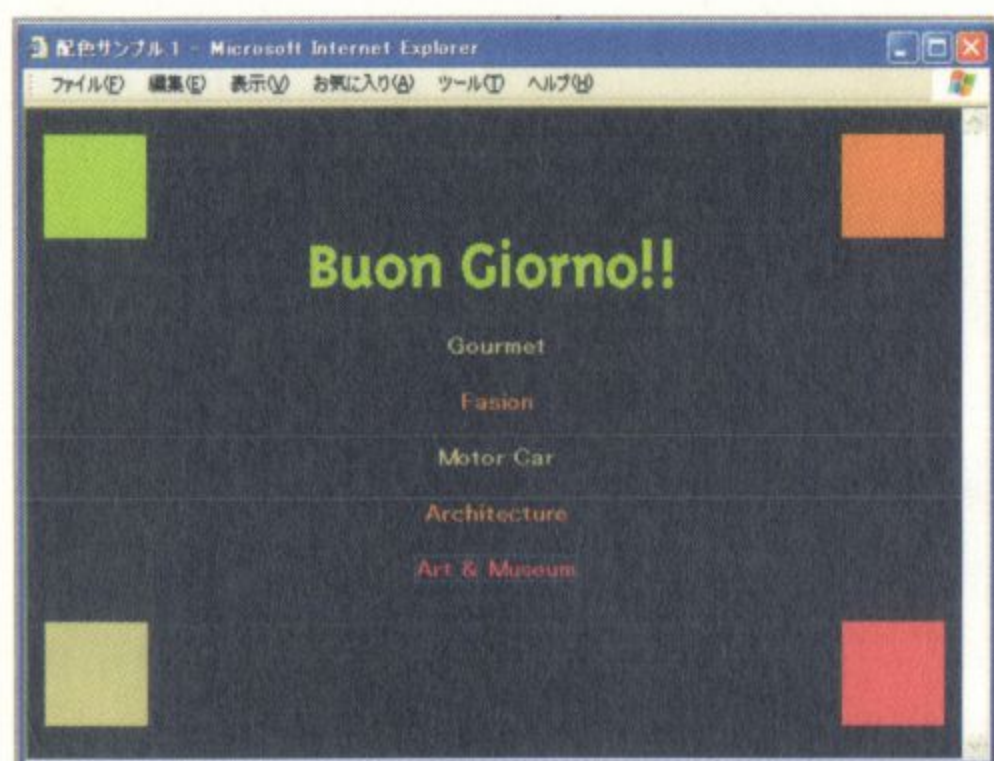
#0066ff	#ccff00
#0000ff	#66ff00
#9900ff	#00ff33
#ff00ff	#00ff99

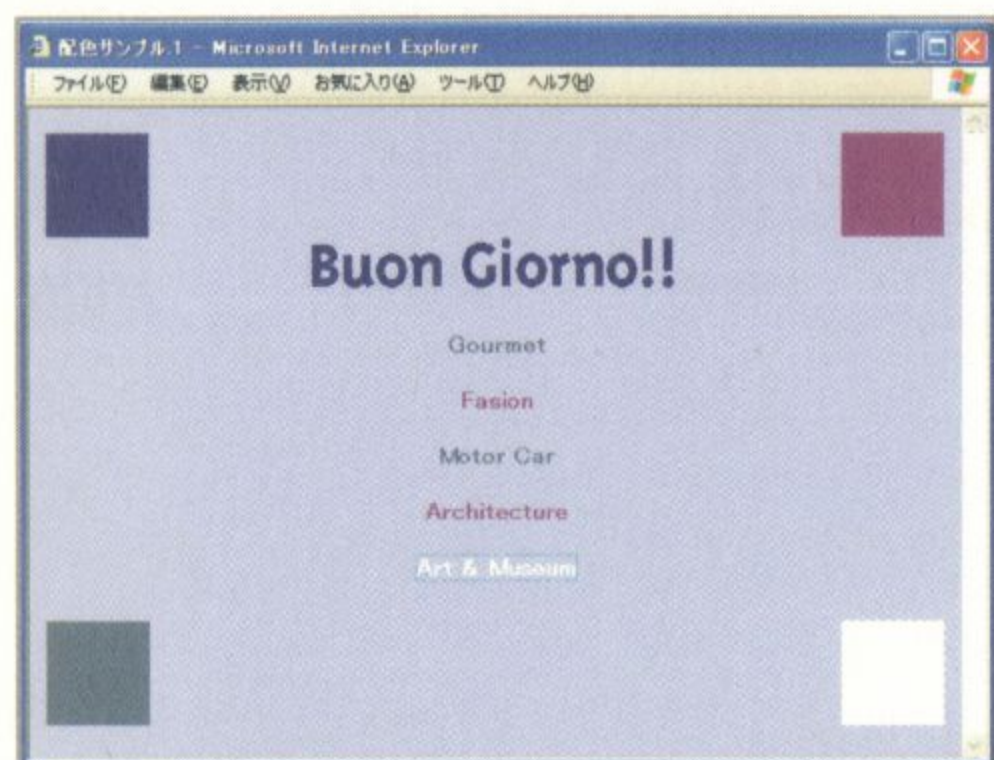
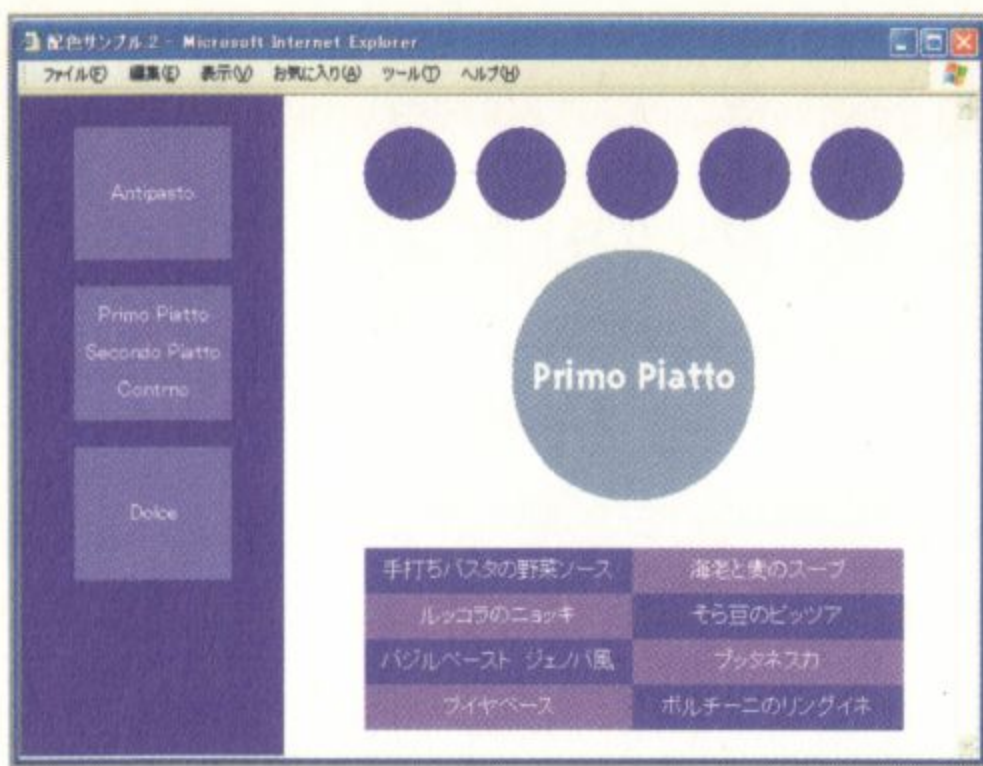
#0066ff	#ff0000
#0000ff	#ff6600
#9900ff	#ffff00
#ff00ff	#99ff00

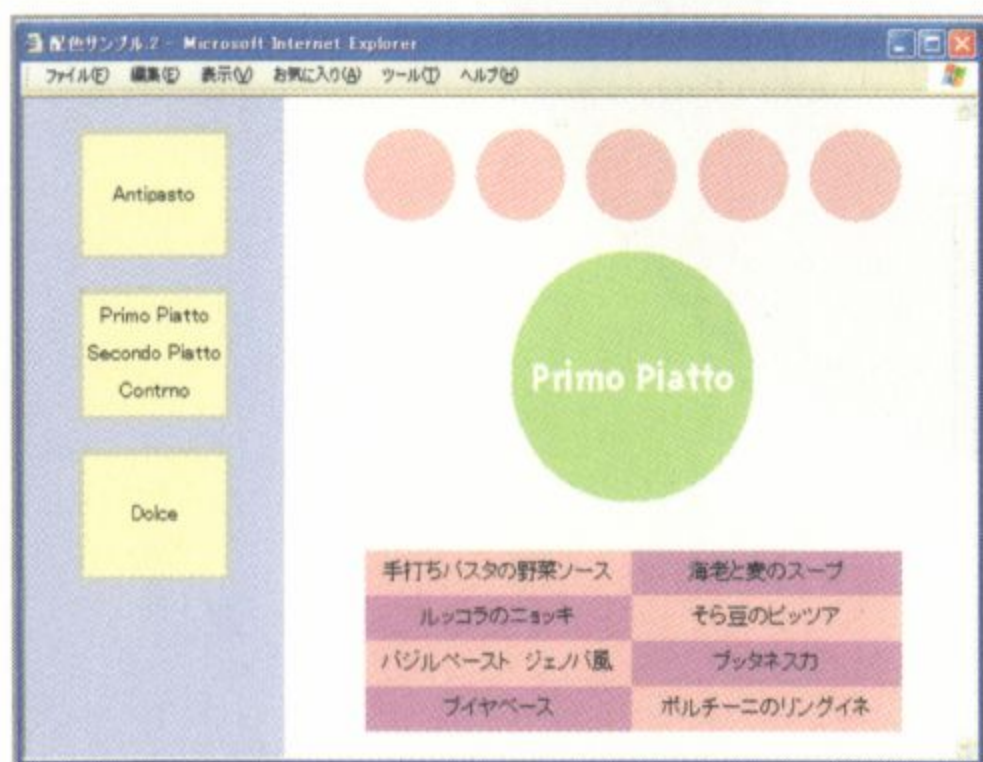
明度のバリエーション

配色サンプル1 - Microsoft Internet Explorer
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#9966cc bgcolor
#ccffff text
#ccff33 link
#99cccc vlink
#ffcc33 alink

配色サンプル1 - Microsoft Internet Explorer
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#663399 bgcolor
#ffcccc text
#ff6699 link
#cc9999 vlink
#ffff99 alink

配色サンプル1 - Microsoft Internet Explorer
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#330066 bgcolor
#ccff66 text
#ff9933 link
#cccc99 vlink
#ff3366 alink

配色サンプル1 - Microsoft Internet Explorer
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#ccccff bgcolor
#003399 text
#990099 link
#666699 vlink
#ffffff alink

▲白地によって、すっきりした印象になります

▲紫系の同系色で明度差のない配色は、重苦しい感じになります

▲高明度の配色は、広い色相を取り入れやすくなります

▲純色に近い色調で広い色相を用いると、雑然とした印象になりやすくなります

● 濃暗色系の配色

濃暗色は、彩度や明度が低い色です。彩度・明度によって「ダーク系」「ダル系」「グレイッシュ系」などに分類されます。濃暗色系の中でも、彩度・明度が比較的高い、ややくすんだ感じの色は、自然界で目にすることの多い色調であるため、「アースカラー」とも呼ばれます。

濃暗色の配色は、一般に重く鈍い印象を与えます。主に男性的で年齢層の高い印象の色で、重厚・渋み・伝統などのイメージを伝える場合には欠かせません。一方、配色によっては、暗い・寂しい・地味などのマイナスイメージを作ることにもなります。また、彩度・明度が低くなるほど無機的な印象になるため、モノトーン系の性格も含むようになります。

同系色の配色

#663333	#660000	#663300
#666633	#663300	#006600
#336666	#006666	#003366
#333366	#330066	#660066

#993333	#990000	#996600
#669966	#339900	#009933
#669999	#009999	#003399
#996699	#660099	#990066

補色・準補色との対比

#660000	#006666
#663300	#330066
#006666	#660000
#330066	#663300

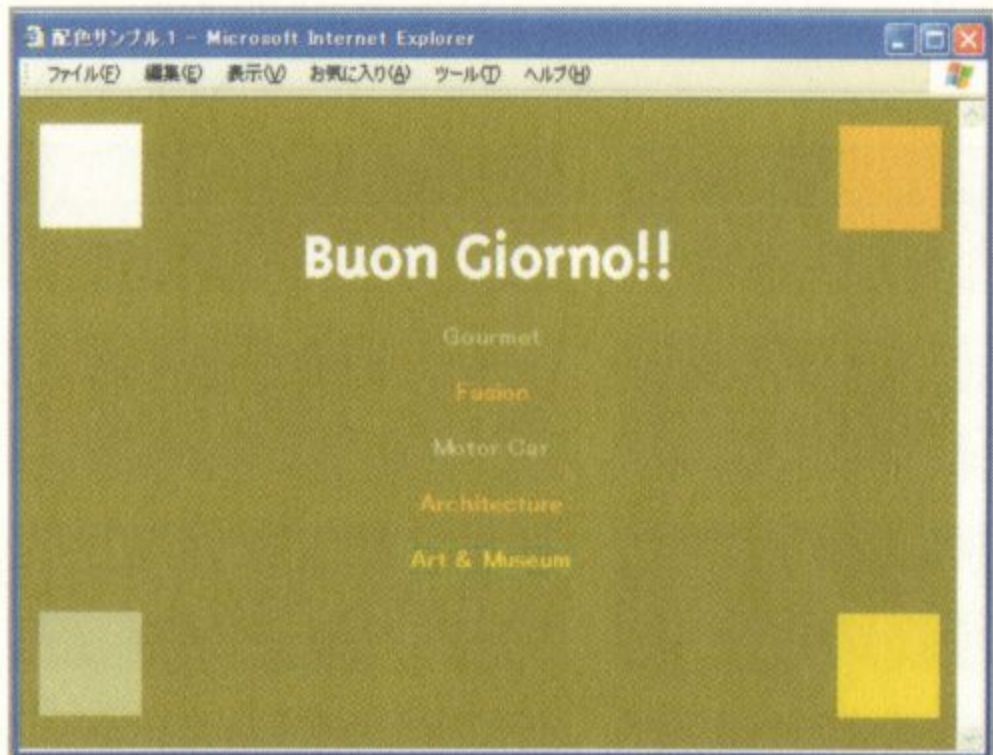
#660000	#006633
#663300	#000066
#006666	#660033
#330066	#666600

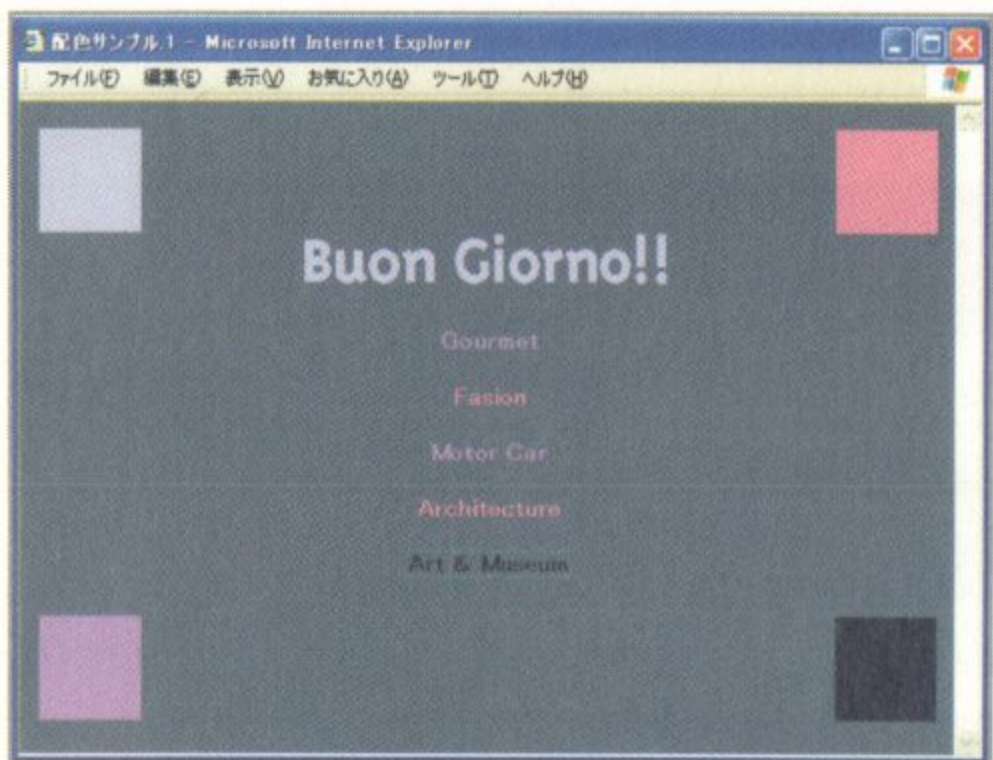
#660000	#003366
#663300	#660066
#006666	#663300
#330066	#006600

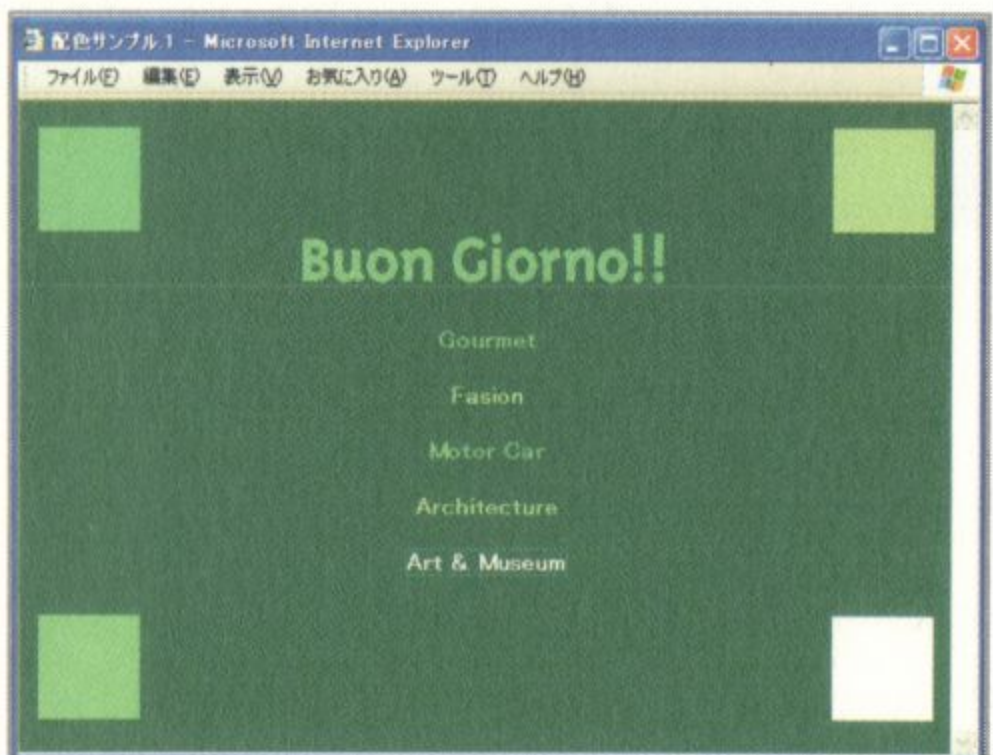
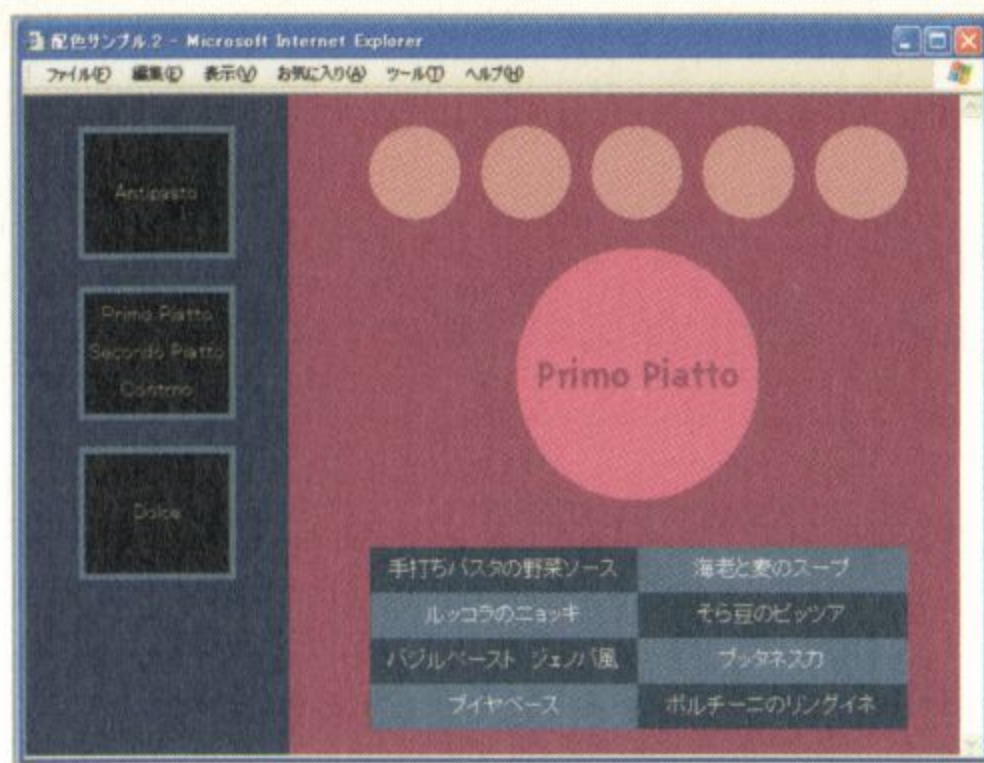
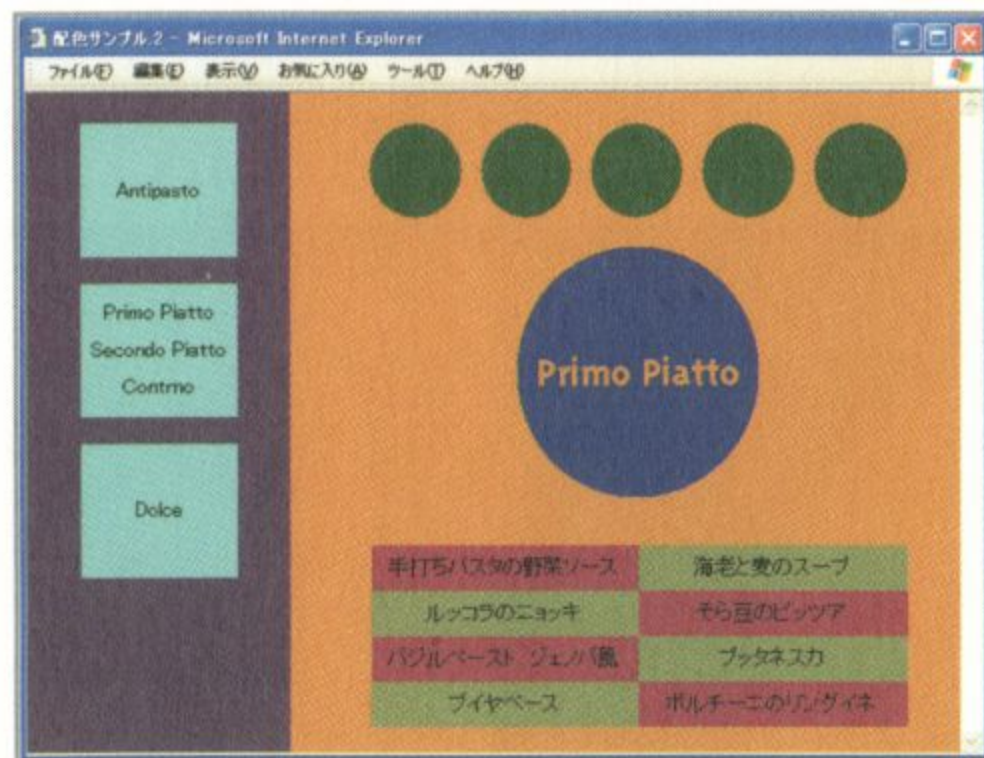
明度のバリエーション

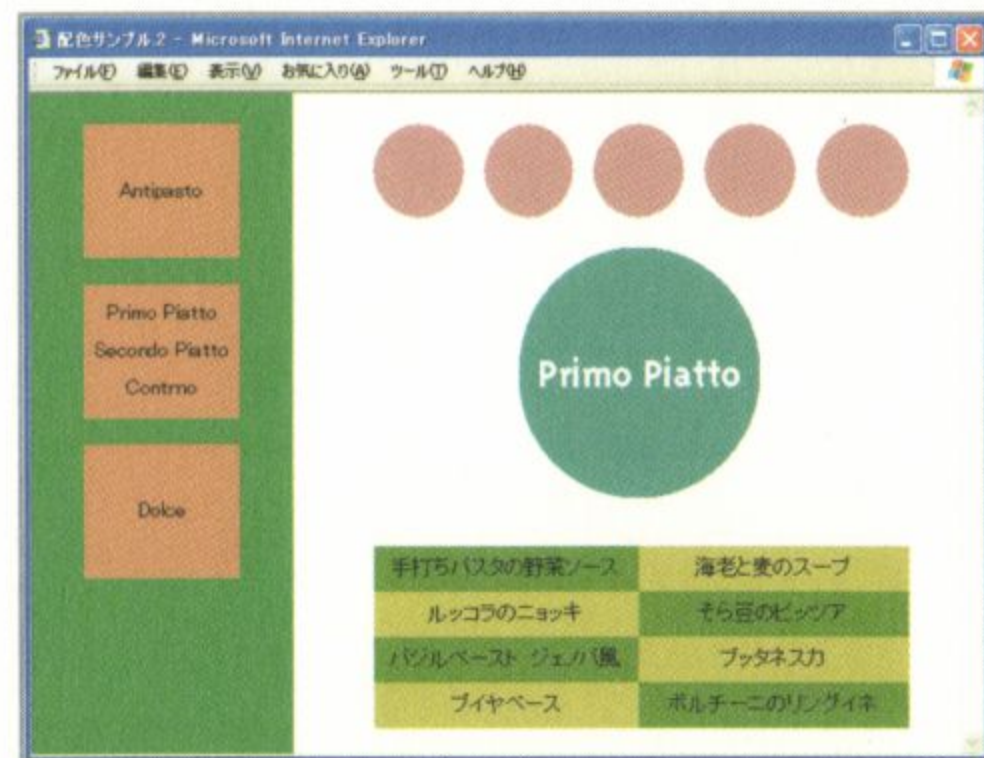
配色サンプル1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#660000 bgcolor
#ffcccc text
#ff6666 link
#ff9999 vlink
#ccff66 alink

配色サンプル1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#999933 bgcolor
#ffffff text
#ffcc00 link
#cccc99 vlink
#ffff00 alink

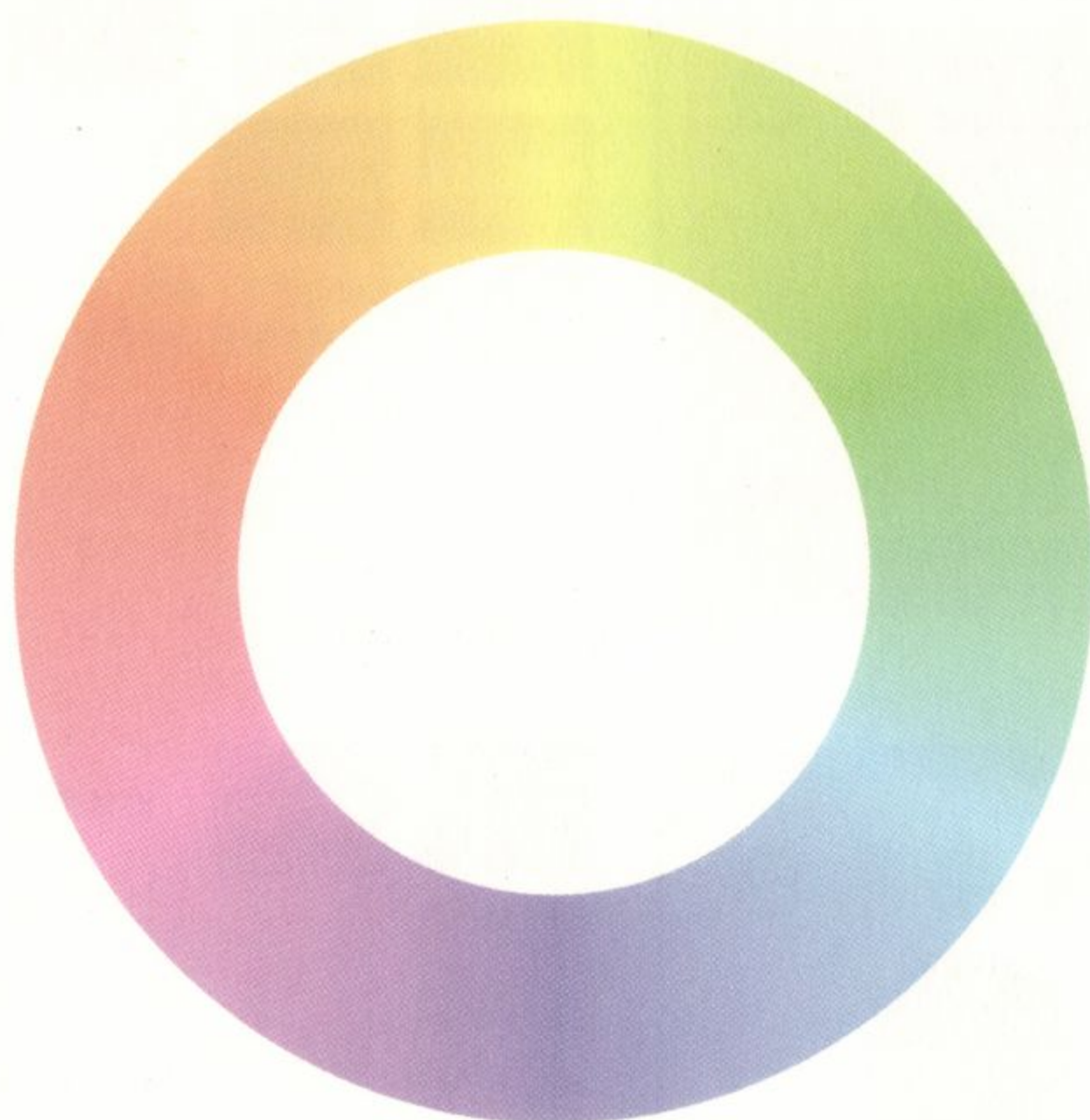
配色サンプル1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#666699 bgcolor
#ccccff text
#ff99cc link
#cc99ff vlink
#330066 alink

配色サンプル1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#336666 bgcolor
#66ff99 text
#ccff99 link
#99cc99 vlink
#ffffff alink

▲彩度が高く、明度が低い配色は、円熟した雰囲気を作ります

▲高彩度・低明度の色を、補色に近い対比で組み合わせると、和風またはエスニック風の配色になります

▲青系の暗色には、一般に男性的なイメージがあります

▲ややくすんだ色調のアースカラーによる配色は、好感度が高くやわらかい印象を与えます

● 淡明色系の配色

淡明色系は、濃暗色系とは反対に、彩度や明度が高い色です。彩度・明度によって「ライト系」「ペール系」「ライトグレイッシュ系」などに分類されます。淡明色系の中で彩度・明度が比較的低い色も、「アースカラー」に含まれます。

淡明色系の配色は、軽やかで柔らかい印象を与え、女性に好まれる色調です。明度が非常に高く白に近いベージュなどの色は、個性は少ないものの、安心感があって受け入れられやすく、上品でやさしい印象を作ることができます。一方、淡明色だけでコントラストの少ない配色は、弱々しくあいまいな印象にもなります。

同系色の配色

#ffffcc	#ffff99	#ccff99
#ccffcc	#99ffcc	#99ffff
#ccccff	#9999ff	#cc99ff
#ffccff	#ff99cc	#ff9999

#66ff66	#99ff99	#99ffcc
#6699ff	#99ccff	#99ffff
#ff66ff	#ff99ff	#ff99cc
#ff9966	#ffcc99	#ffff99

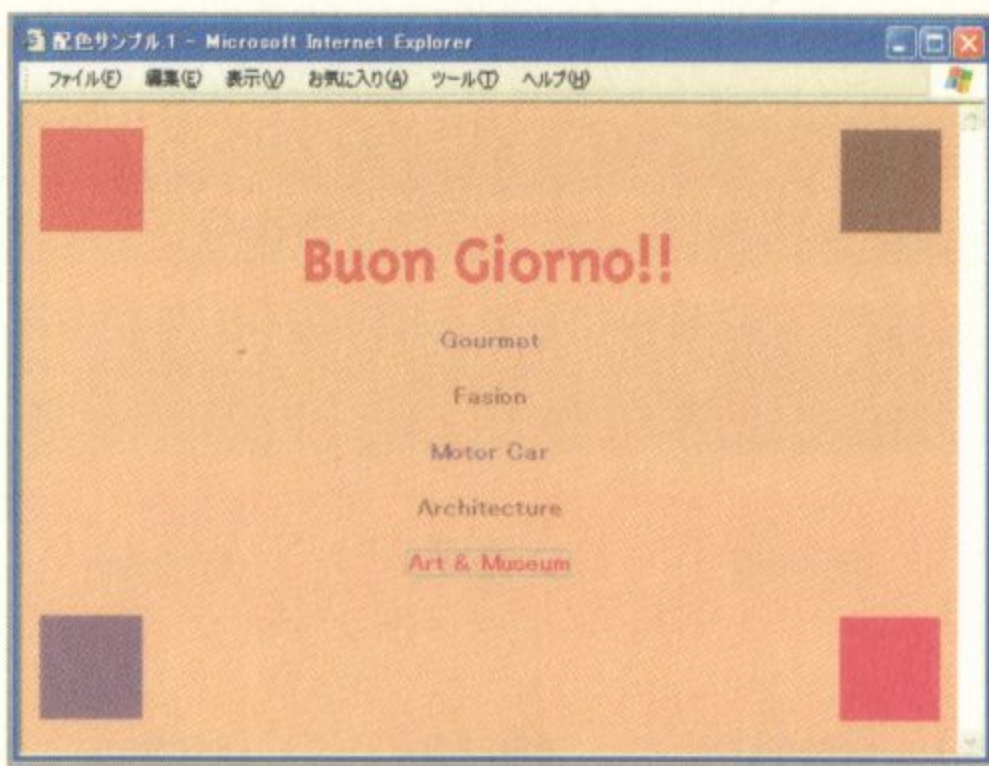
補色・準補色との対比

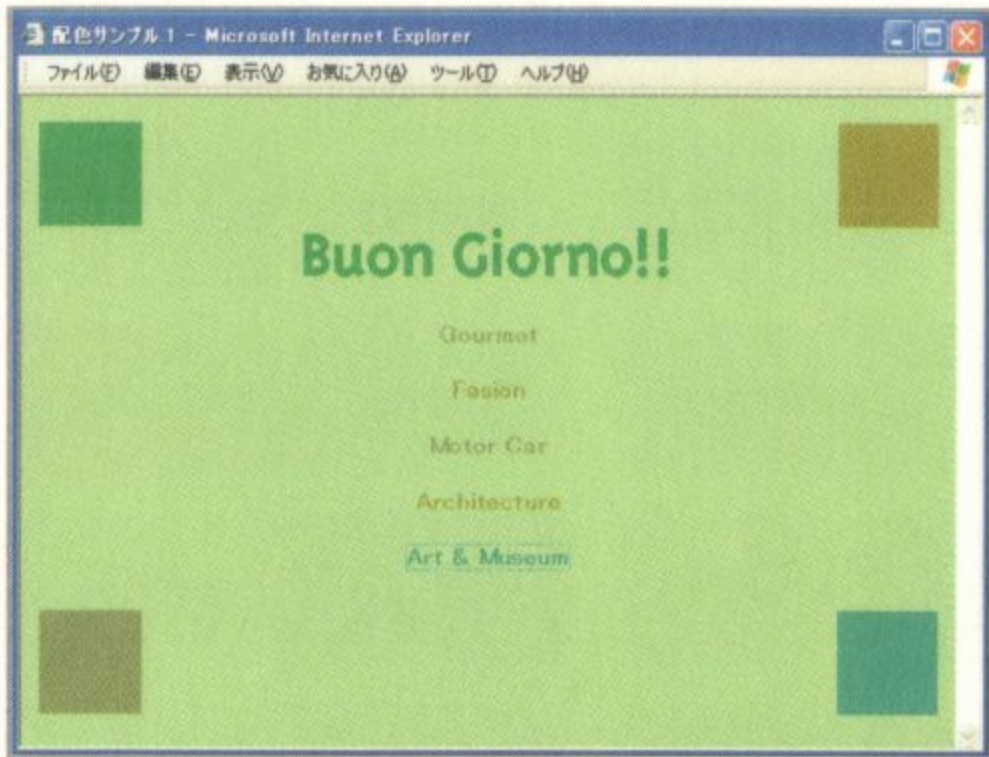
#ffff99	#9999ff
#99ffcc	#ff99cc
#9999ff	#ffff99
#ff99cc	#99ffcc

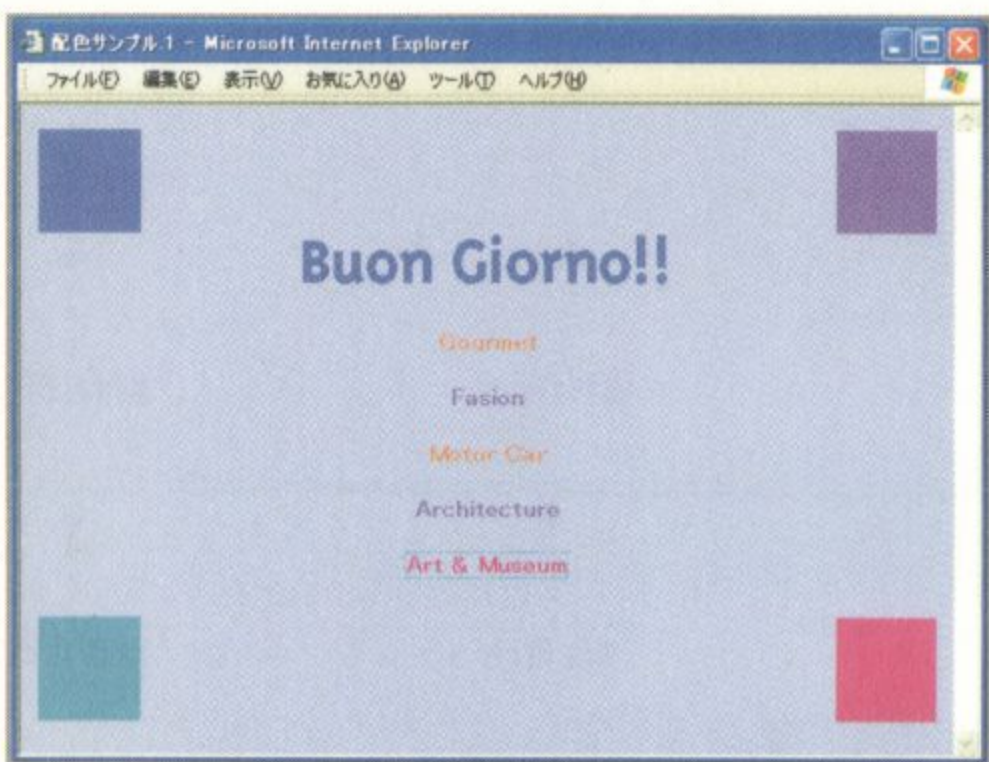
#ffff99	#99ccff
#99ffcc	#ff99ff
#9999ff	#ffcc99
#ff99cc	#99ff99

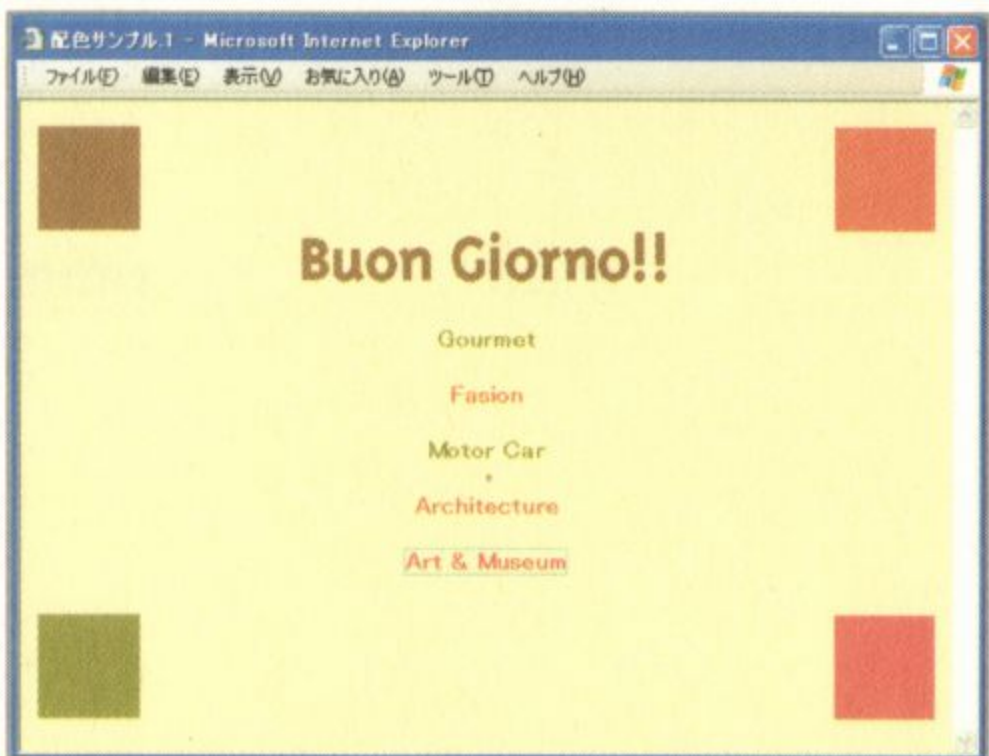
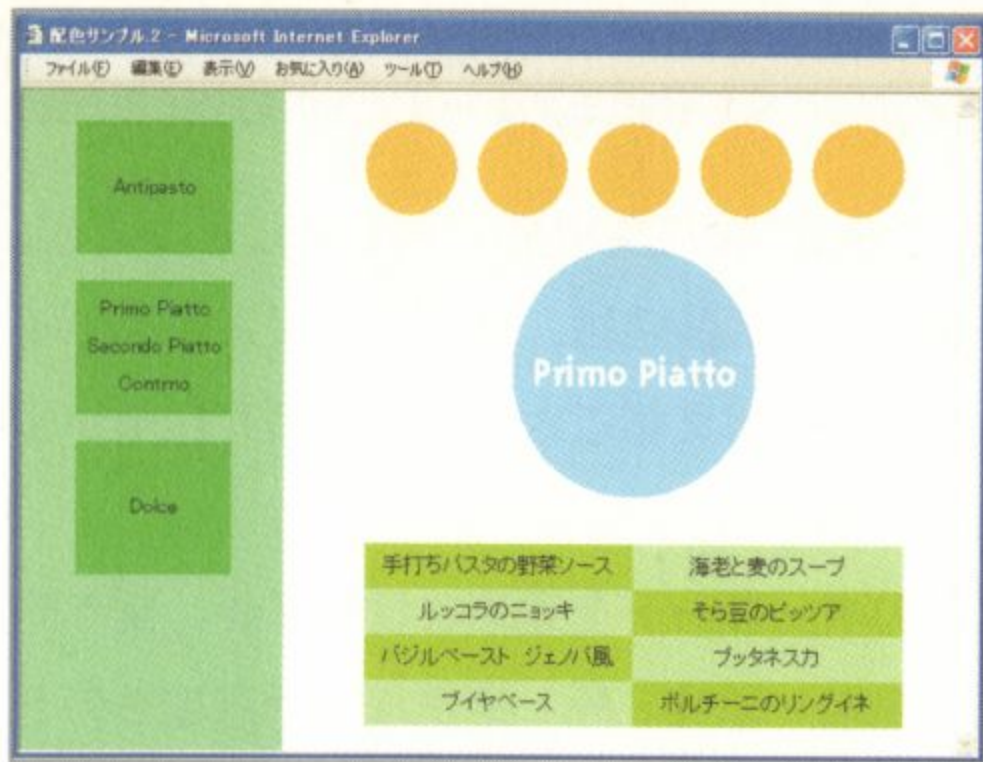
#ffff99	#cc99ff
#99ffcc	#ff9999
#9999ff	#ccff99
#ff99cc	#99ffff

明度のバリエーション

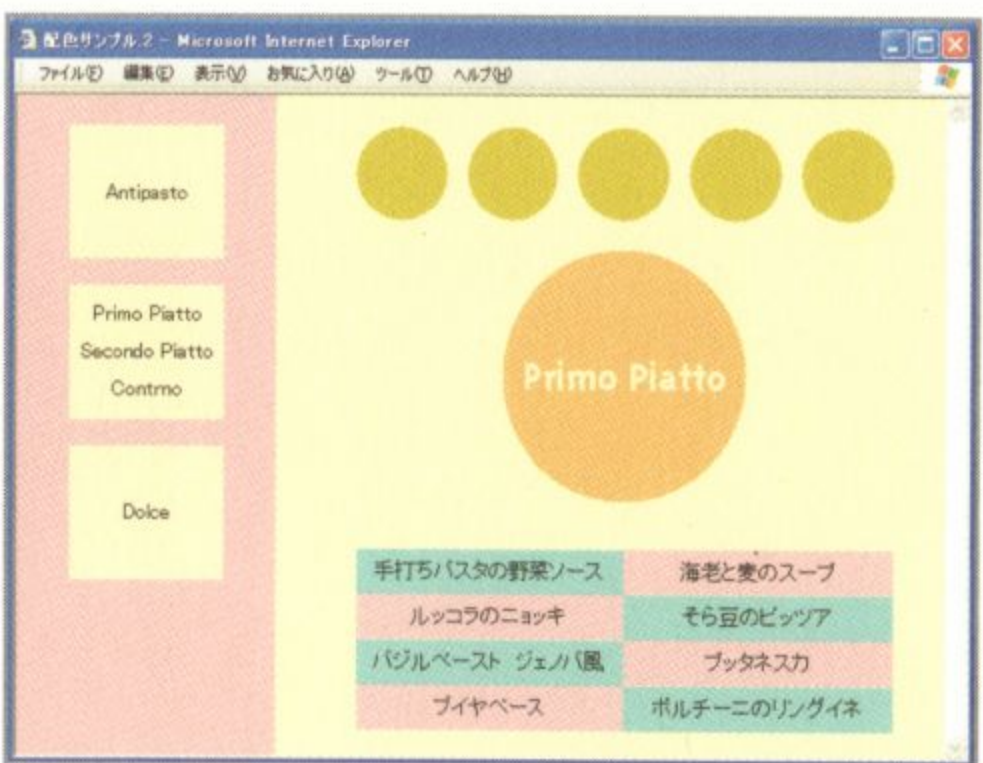
配色サンプル.1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#ffcc99 bgcolor
#cc6666 text
#996666 link
#996699 vlink
#cc3366 alink

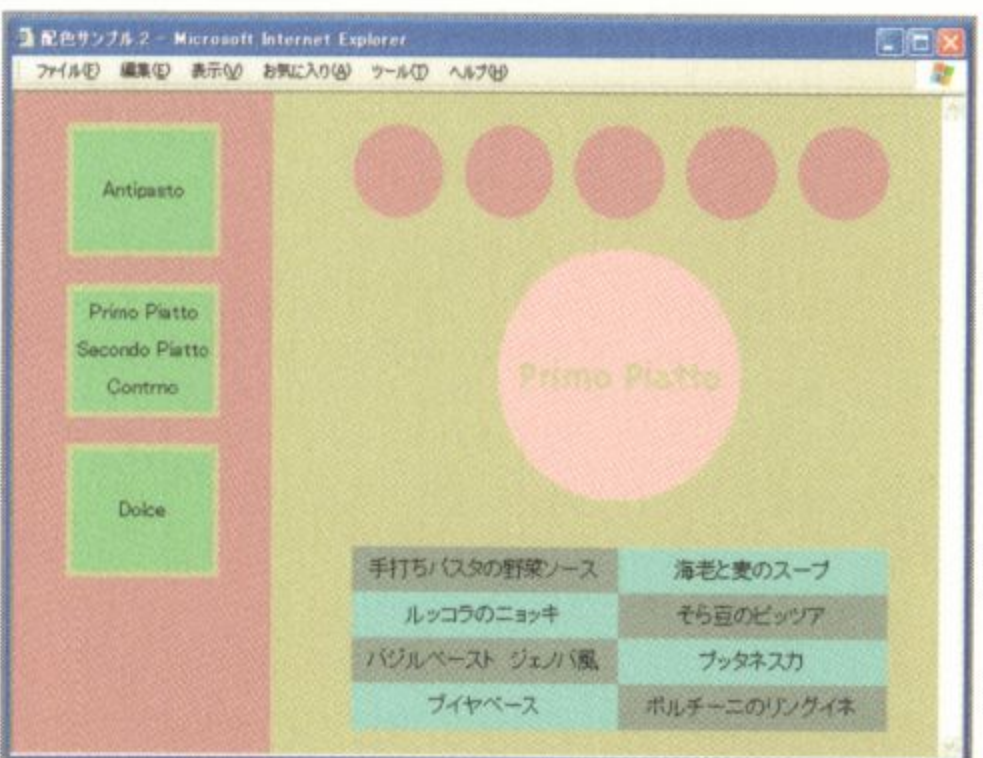
配色サンプル.1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#ccff99 bgcolor
#339966 text
#999900 link
#999966 vlink
#009999 alink

配色サンプル.1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#ccccff bgcolor
#6666cc text
#9966cc link
#6699cc vlink
#cc0099 alink

配色サンプル.1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#ffffcc bgcolor
#996600 text
#cc6633 link
#999933 vlink
#ff0033 alink

▲彩度・明度の高い色と白を組み合わせると、若々しい印象になります

▲彩度が低く、明度の高い配色は、やわらかく落ち着いた印象になります

#ffcccc #ffffcc #cccc00 #ffcc66 #99cccc

▲淡明色系の配色は、明るく女性的なイメージを作ることができます

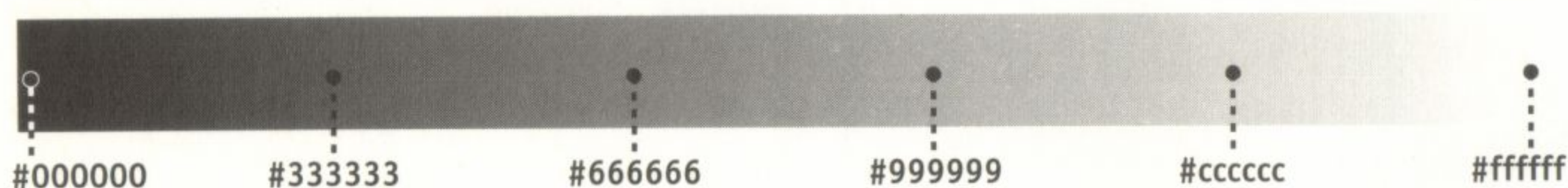
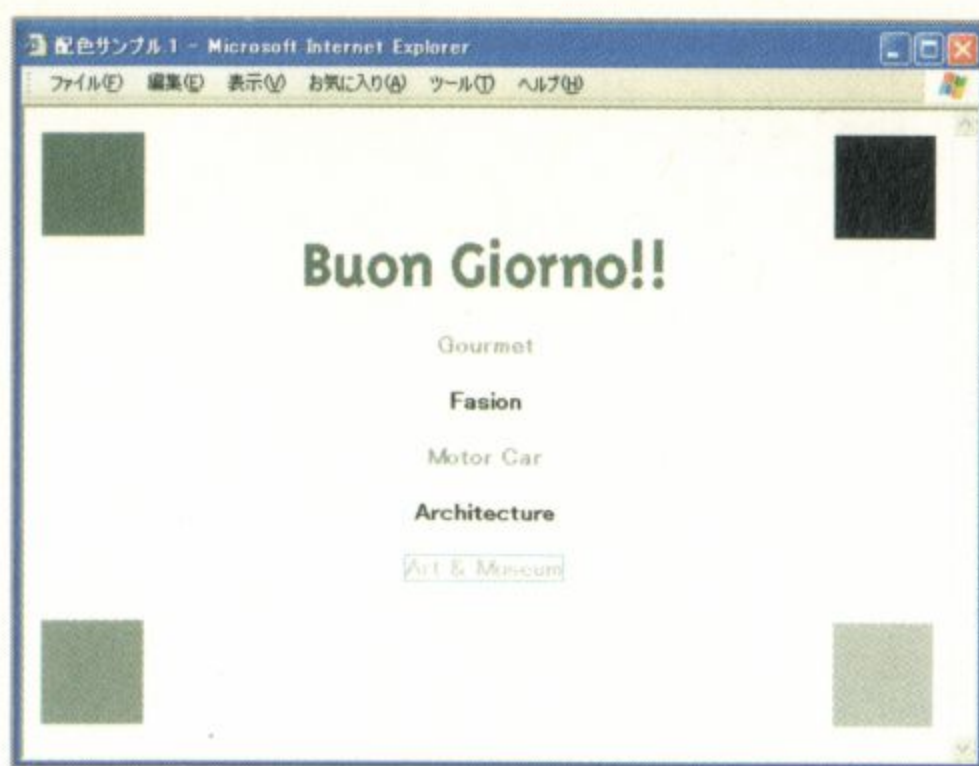
#cc9999 #cccc99 #99cc99 #ffcccc #999999
#99cccc

▲彩度が低く、明度差の少ない配色は、あいまいでのんびりとした雰囲気になります

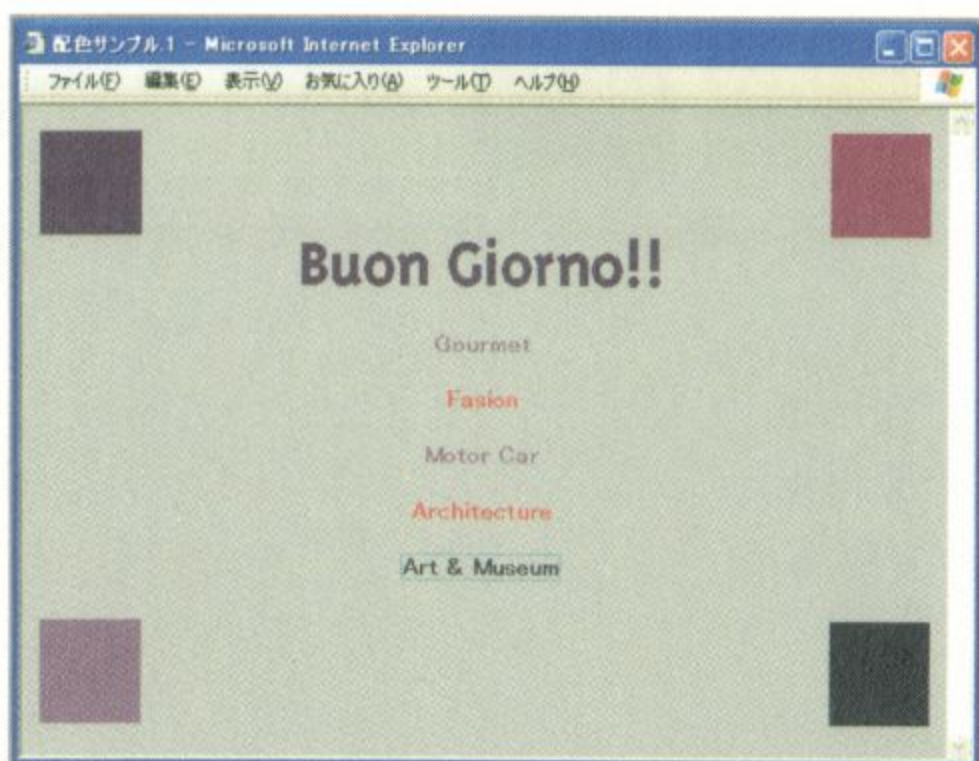
● モノトーンの配色

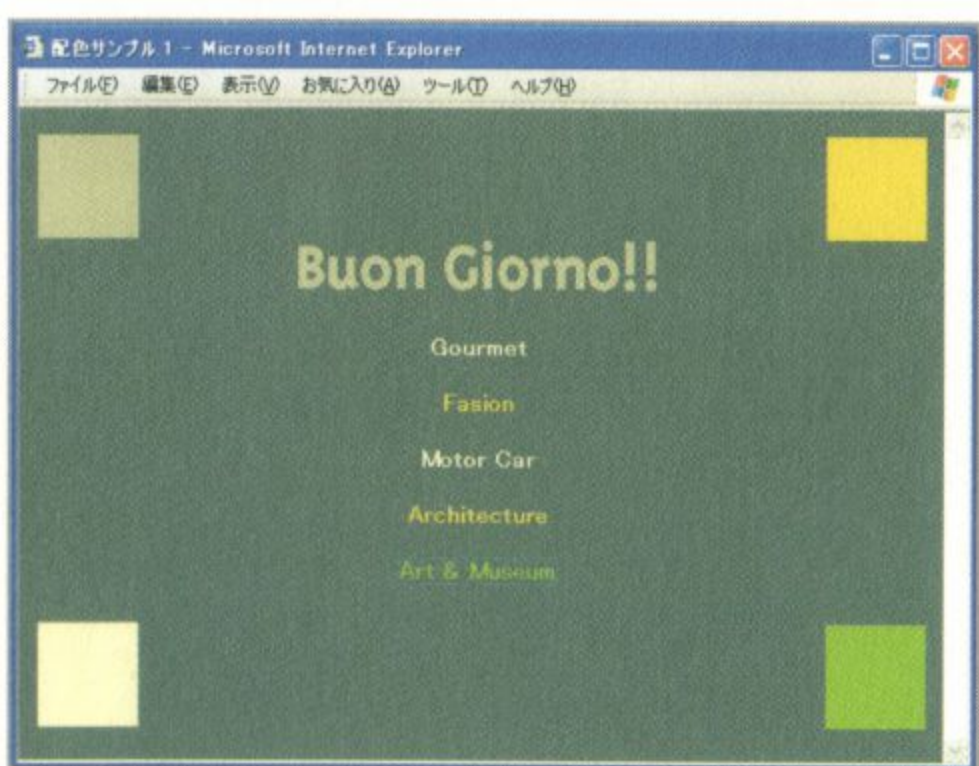
無彩色の白・黒・グレーはニュートラルな色なので、どんな色とでも組み合わせることができ、配色によって印象が変わります。有彩色を加えないモノトーンの配色は、モダンで大人っぽいイメージになりますが、バランスによっては暗く寒々しい印象を与えることにもなるので注意が必要です。

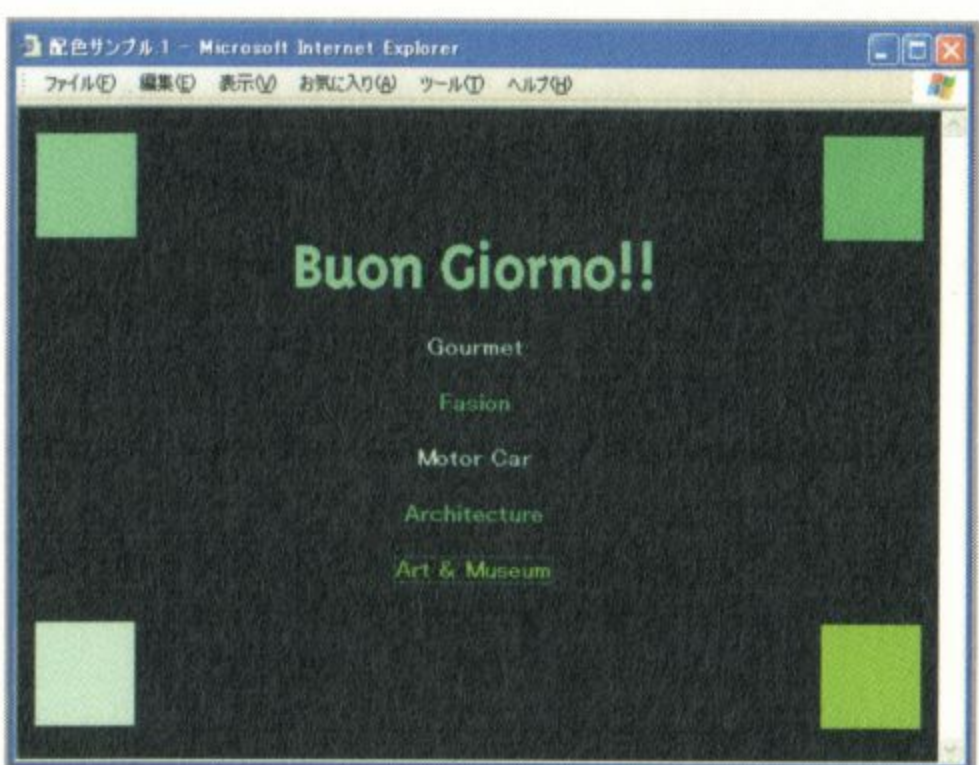
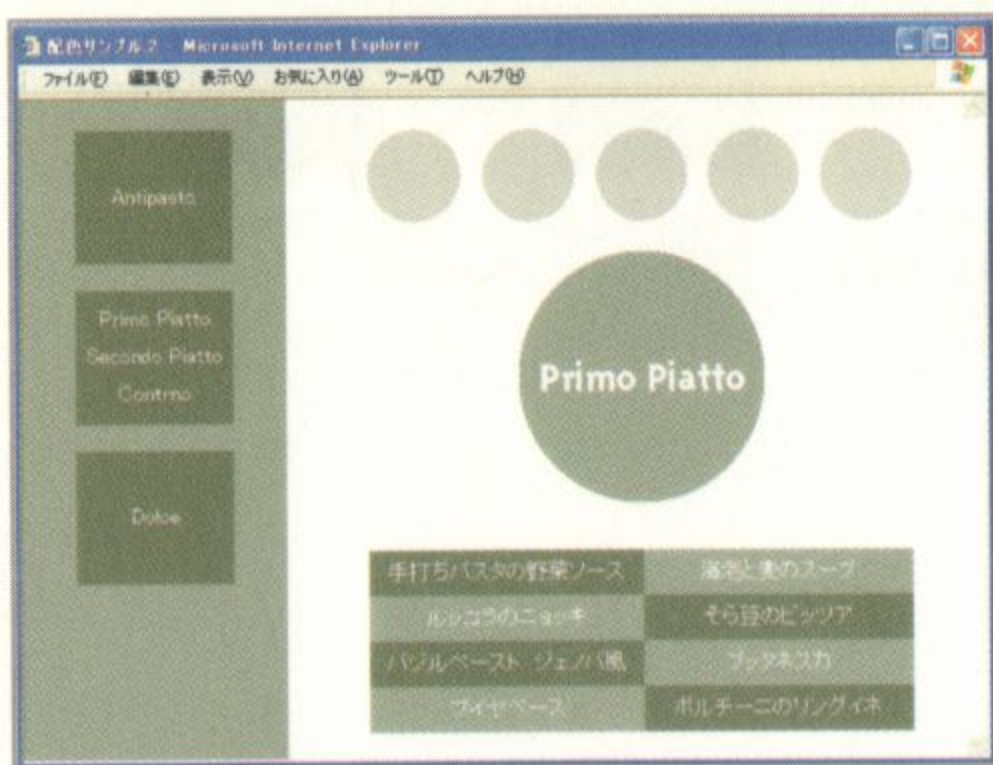
単色の場合、白は清潔・清楚・穢れがない・儚いなどのキーワードを連想させますが、基本的にマイナスイメージは少ない色です。反対に、黒は夜・暗闇・恐怖・死・絶望など不吉なものを象徴する一方、洗練されてシャープな印象を与える色でもあります。グレーもシックで落ち着いた印象の色ですが、使い方によっては地味で陰気なイメージとなります。

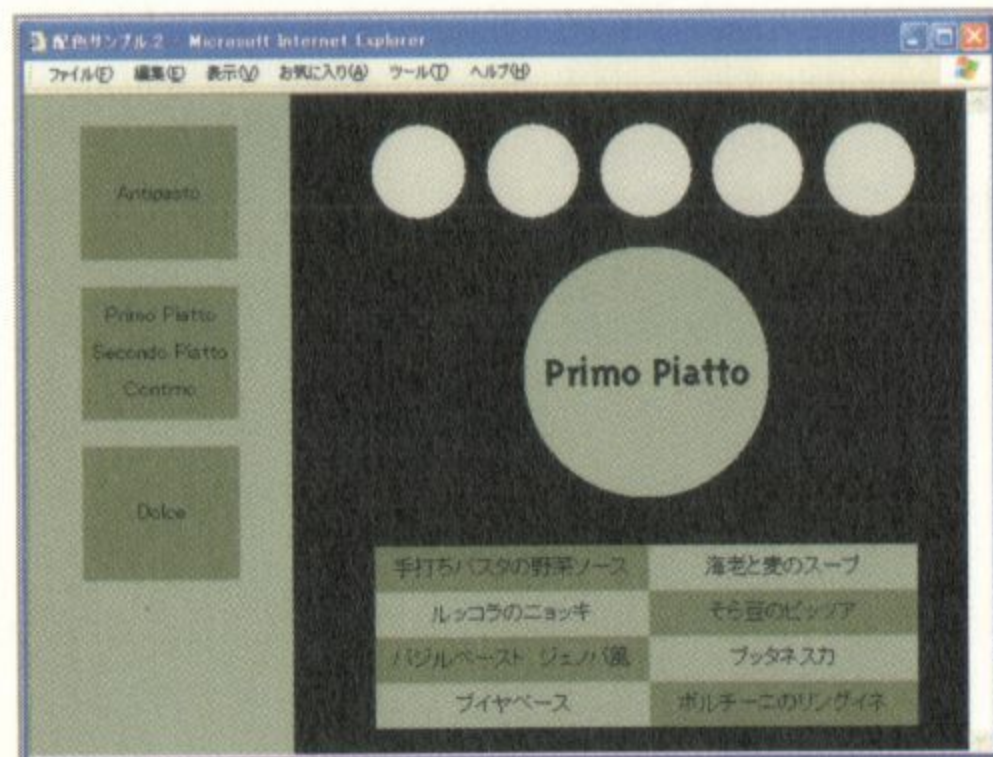
明度のバリエーション

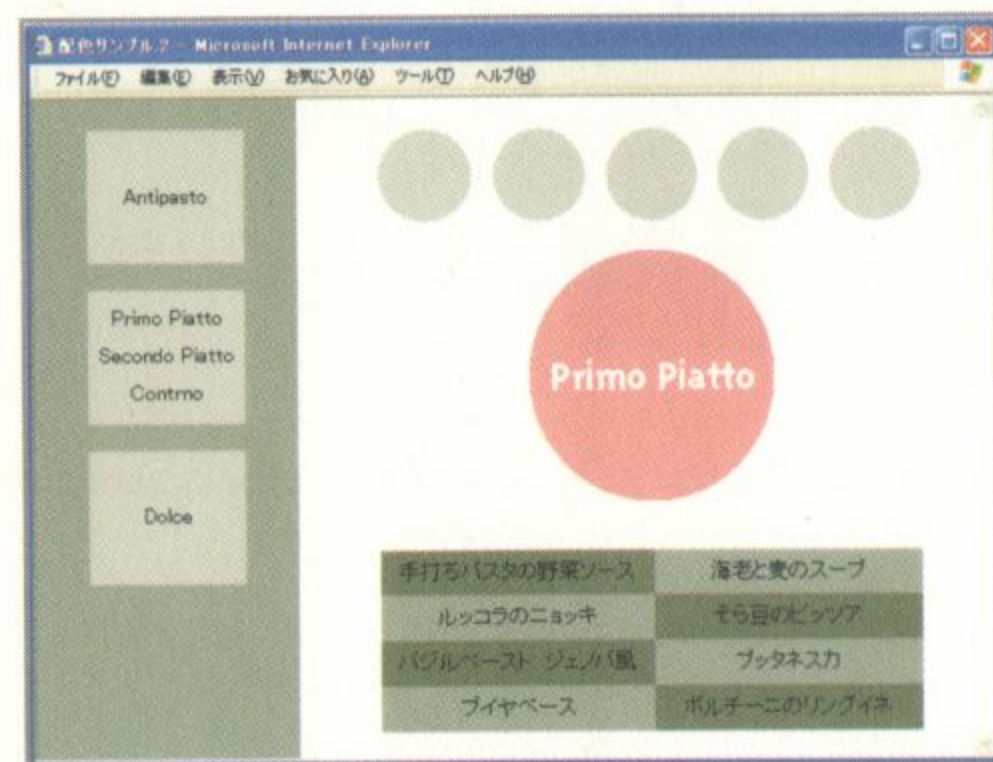
配色サンプル.1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#ffffff bgcolor
#666666 text
#000000 link
#999999 vlink
#cccccc alink

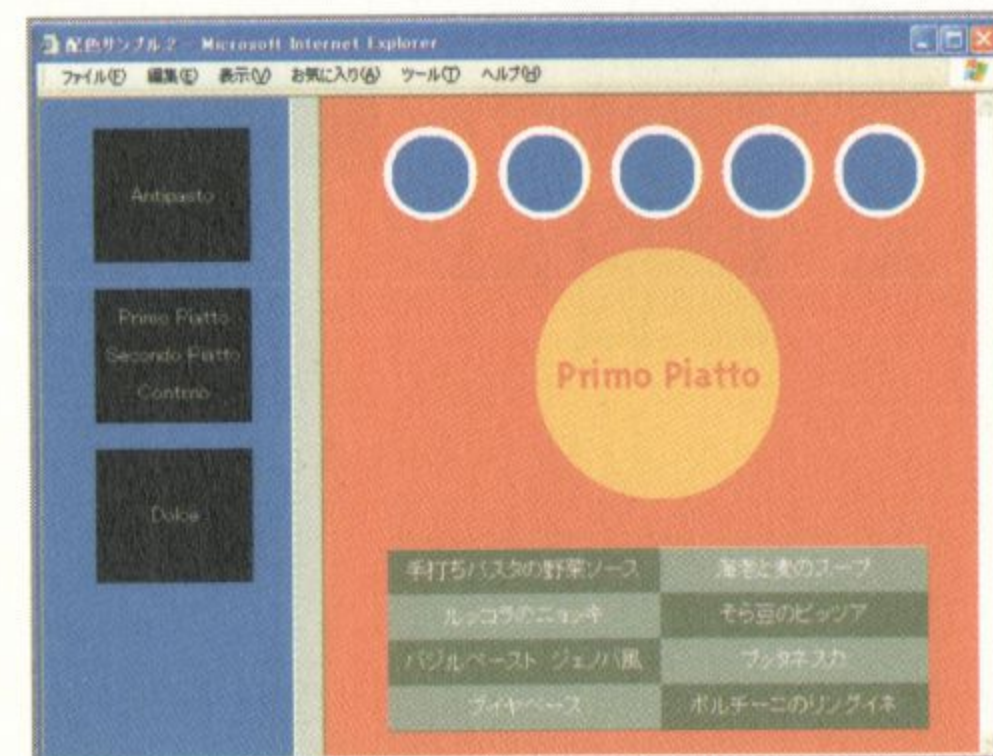
配色サンプル.1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#cccccc bgcolor
#660066 text
#990066 link
#996699 vlink
#330033 alink

配色サンプル.1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#666666 bgcolor
#cccc99 text
#ffff00 link
#ffffcc vlink
#99cc33 alink

配色サンプル.1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
Buon Giorno!!
Gourmet
Fasion
Motor Car
Architecture
Art & Museum

#000000 bgcolor
#99cc99 text
#66ff66 link
#ccffcc vlink
#99cc33 alink

#999999 #666666 #ffffff #cccccc

▲無彩色による配色は、寒々しい印象になることがあります

#999999 #666666 #000000 #cccccc

▲黒を基本色とすると、強い主張が感じられるようになります

#999999 #cccccc #ffffff #ff9999 #666666

▲モノトーンの配色は、有彩色との組み合わせによって印象が変化します

#3366cc #000000 #cccccc #ff6633 #ffffff
#ffcc66 #666666 #999999

▲補色の間に無彩色を置くと、すっきりした対比になります

4 APPENDIX

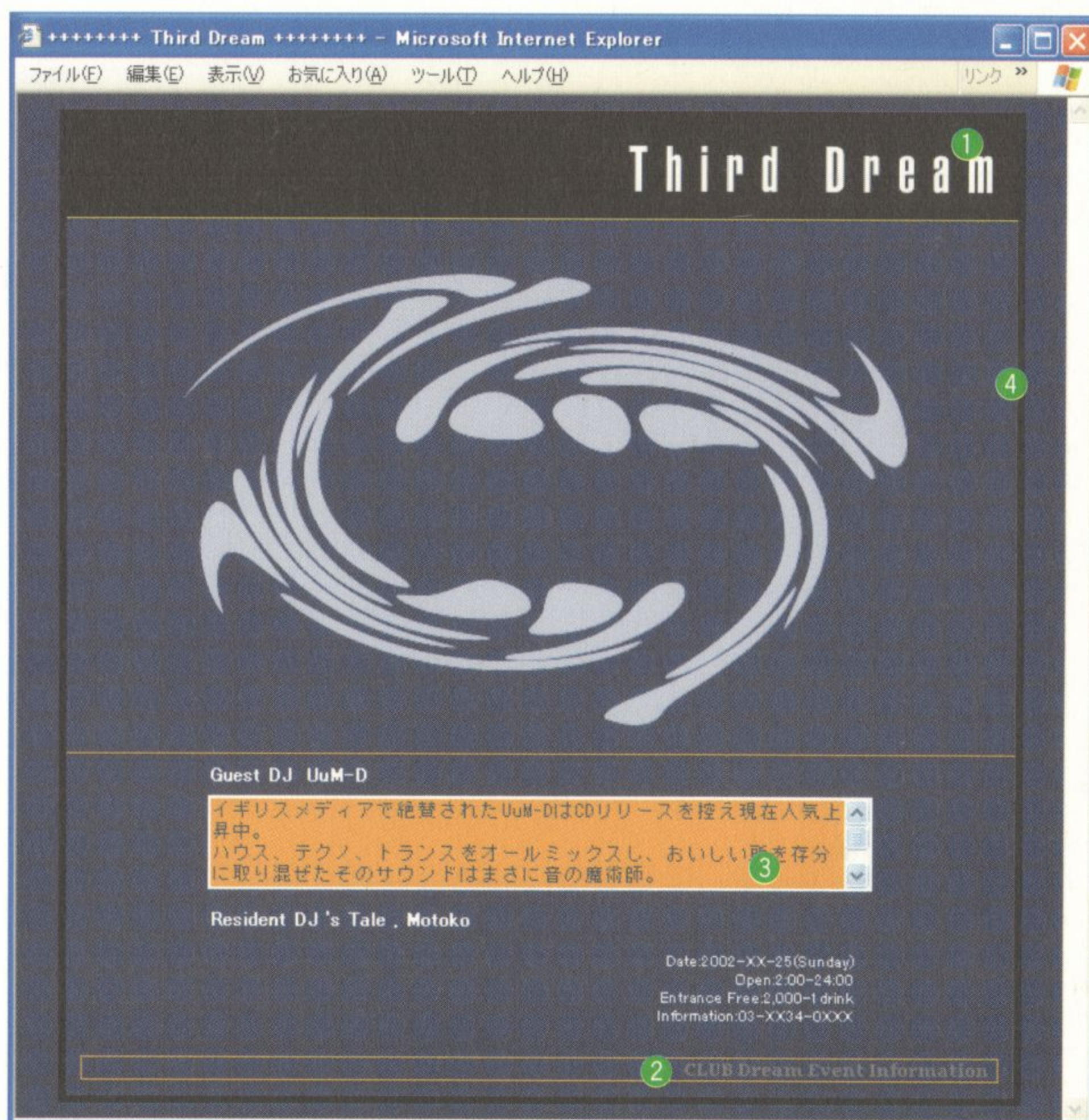
ビジュアルインデックス

ビジュアルインデックスでは、本書に掲載しているスタイルシートを利用したサンプルページをご紹介します。

実際のWebページでは、未対応ブラウザへの配慮やブラウザやバージョンによって生じるバグ（不具合）の回避策なども考慮に入れる必要があります。ここでは実例として各種のバグ回避策も講じていますので参考にしてください。

● スタイルシートを内部記述したページ

HTML文書内の冒頭部分に、<style>タグでスタイルシートを設定するサンプルです。本書リファレンス部分のサンプルソースはこの形式で記述されています。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">          p.51
<html lang="ja">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv=Content-Style-Type content="text/css">                    p.10
<title>++++++++ Third Dream ++++++++</title>
<style type="text/css">                                                    p.15
<!--
body            {
    margin: 10px 28px;
    padding: 0px;
    background: url("dot.gif") #006 repeat;                                p.157
    color: #fff
}
❶ h1            {
    margin-top: 0px;
    padding: 15px;
    background: #000;
    color: #fff;
    text-align: right;
    font-size: large;
    font-family: Georgia, "Times New Roman", Times, serif;
    border-bottom: #f93 1px solid                                          p.192
}
❷ h2            {
    margin: 10px;
    padding-right: 5px;
    background: transparent;
    color: #666;
    text-align: right;                                                     p.69
    font-size: x-small;                                                    p.117
    font-family: Georgia, "Times New Roman", Times, serif;
    border: #f93 1px solid
}
form            {
    margin: 0px;
    padding: 5px
}
❸ textarea      {
    width: 450px;
    background: #f93;                                                      p.157
    color: #006;                                                           p.56
    font-size: x-small;
    border: 2 solid #fff            /* 【注意1】 */
}
#MainImage      {                                                          p.25
    text-align: center;
    background: transparent;
    color: #fff;
    margin-top: 10px
}
#MainStyle      {
    text-align: center;
    margin-top: 10px;
    border-top: #f93 1px solid
}
❹ div.Over      {                   /* 全体を枠線で囲うためのスタイル「Over」 */
```

```
        border-style: solid;                                          p.188
        border-width: 5px;                                            p.174
        border-color: #000                                            p.182
}
div.Special      {                                                    p.25
        background: transparent;
        color: #fff;
        margin-top: 8px;
        margin-right: 15%;
        margin-left: 15%;
        text-align: left;
        font-weight: bold;
        font-size: small
}
div.Info         {
        background: transparent;
        color: #fff;
        margin-top: 5px;
        margin-right: 15%;
        margin-left: 15%;
        padding: 10px;
        text-align: right;
        font-weight: normal;
        font-size: x-small
}
-->
</style>
</head>
<body lang="ja">

❹ <div class="Over">                   <!--=====全体を枠線で囲うスタイル「Over」適用=====-->

❶ <h1><img src="fryertitle.gif" width="250" height="36" alt="Third Dream"></h1>
<div id="MainImage"><img src="fryer.gif" width="500" height="323" alt=""></div>

<div id="MainStyle">   <!-- =====スタイル「MainStyle」適用===== -->          p.25

<div class="Special">Guest DJ　UuM-D</div>                            p.25
<form>
❸  <textarea rows="4" cols="50">イギリスメディアで……（中略）……体感しよう。</textarea>
</form>
<div class="Special">Resident DJ 's Tale , Motoko</div>

<div class="Info">
Date:2002-XX-25(Sunday)<br>Open:2:00-24:00<br>Entrance Free:2,000-1drink<br>Information:03-XX34-0XXX<br>
</div>

</div>                   <!-- =====スタイル「MainStyle」適用ここまで===== -->

❷ <h2>CLUB Dream Event Information</h2>

</div>                   <!--=====全体を枠線で囲うスタイル「Over」適用ここまで=====-->

</body>
</html>
```

【注意1】本来はborder: 2px solid #fffのように単位を書かなければ構文的にエラーですが、Netscape Navigator 4.xではフォームの要素にこれを指定すると表示がくずれるというバグがあります。その対策としてここでは単位を付けずに指定する方法をとっています。

● スタイルシートを外部記述したページ

フォームのHTML文書に、外部ファイルに記述したスタイルシートを読み込むサンプルです。<head>タグ内部の<link>タグで読み込む外部ファイルを指定します。

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv=Content-Style-Type content="text/css">
<link rel="stylesheet" href="seminar.css">
<title>次世代BtoC戦略セミナー</title>
</head>

❶ <body lang="ja">

❷ <h1>次世代BtoC戦略セミナー</h1>
❸ <h2>あなたの会社はBtoCを100%活用していますか？</h2>

❹ <div class="Main">                <!-- =====スタイル「Main」適用===== -->

❺ <img src="photo.jpg" width="237" height="274" alt="BtoC戦略セミナー" class="photo">
<p>
現在は頭打ちといわれているBtoC市場ですが、米国では衣料業界が旧態然とした「オンラインカタログ」
から、利用ユーザーの情報を最大限に取り入れ、個々の要求に対応できるシステムを打ち出すなどして成
功を収めています。<br>
特に、扱う商品の性質から、ネット利用者のうち、半数を占める20代後半から30代前半の層にターゲッ
トを絞り、なかでも女性を重視した戦略に踏み込んだことがより多くのユーザーをオンラインショッピン
グの顧客として取り込む鍵になったと思われます。<br>
日本でのネット利用者も、2年前までは約2割だった女性層が半数を占めるまでに伸びて来ていると言わ
れており、そのように素早く柔軟な対応が求められるでしょう。
</p>
<p>
今回のセミナーでは、そのような成功例から、これからの電子商取引におけるビジネスモデルを示し、よ
り深い顧客満足を生むBtoCのノウハウを探って行きます。<br>
プログラム終了後、Eコマースのポータルマスターによる個別の相談も受け付けますので、ぜひご参加く
ださい。
</p>
<p class="txtend">お申込は下記お申込フォームからどうぞ</p>

</div>                <!-- =====スタイル「Main」適用ここまで===== -->

❻ <h3>「次世代BtoC戦略セミナー」講座日程</h3>

<div class="Message">  <!-- =====スタイル「Message」適用===== -->
❼ <table>
  <tr>
    <td>  </td><td class="param">1日目</td><td class="param">2日目</td>
    <td class="param">3日目</td><td class="param">4日目</td><td class="param">5日目</td>
  </tr>
  <tr>
    <td class="param">第1回</td><td>5/24(金)</td>
    <td>5/25(土)</td><td>5/26(日)</td><td>6/1(土)</td><td>6/2(日)</td>
  </tr>
  <tr>
    <td class="param">第2回</td><td>6/11(火)</td>
    <td>6/12(水)</td><td>6/13(木)</td><td>6/14(金)</td><td>6/15(土)</td>
  </tr>
  <tr>
    <td class="param">第3回</td><td>7/5(金)</td>
    <td>7/6(土)</td><td>7/7(日)</td><td>7/13(土)</td><td>7/14(日)</td>
  </tr>
```

```
</table>
</div>                    <!-- ===== スタイル「Message」適用ここまで ===== -->

❻ <h3>「次世代 BtoC 戦略セミナー」お申込 </h3>

<div class="Message">  <!-- ===== スタイル「Message」適用 ===== -->
<form name="form.cgi" method="post" action="">
❽  氏名：<input type="text" name="name1" class="txt"><br>
   性別：<input type="radio" name="radiobutton" value="男">男性
   <input type="radio" name="radiobutton" value="女">女性 <br>
   連絡先：<input type="text" name="address" class="txt"><br>
   希望日：
   <select name="day" size="1">
❾    <option class="color0">お選び下さい</option>
     <option class="color1">第1回：5/24(金)～6/2(日) </option>
     <option class="color2">第2回：6/11(火)～6/15(土)</option>
     <option class="color1">第3回：7/5(金)～7/14(日) </option>
   </select><br>
❿  <input type="submit" name="submit" value="送　信" class="button">
   <input type="reset" name="reset" value="リセット" class="button">
</form>
</div>                    <!-- ===== スタイル「Message」適用ここまで ===== -->

<div class="Navigation"><a href="event.html">イベント情報へ </a></div>

</body>
</html>
```

外部スタイルファイル「seminar.css」

p.383のHTML文書が読み込む外部スタイルファイル「seminar.css」です。

seminar.css

```
❶ body          {
        margin: 0px 0px;
        padding: 0px;
        background: url("bgimg.jpg") #d7d7d7 no-repeat;
        color: #000
    }
❷ h1            {
        margin-top: 0px;
        padding: 15px;
        padding-right: 80px;
        background: url("seminarbg.gif") #36365d no-repeat;
        color: #ddd;
        text-align: right;
        font-size: 14px;
        border-top: #003 2px solid;
        border-bottom: #003 2px solid
    }
❸ h2            {
        margin: 15px 10% 0px 10%;
        padding: 10px 15px 10px 15px;
        background: url("h2bg.gif") #fff no-repeat;
        color: #003;
        text-align: left;
        font-size: 12px;
        border-top: solid 2px #669;        ――― p.192
        border-right: solid 2px #003;      ――― p.192
        border-bottom: solid 2px #000;     ――― p.192
        border-left: solid 2px #666        ――― p.192
    }
❻ h3            {
        margin: 15px;
        margin-left: 15%;                  ――― p.160
        margin-right: 15%;                 ――― p.160
        padding: 10px;
        background: transparent;           ――― p.157
        color: #336;
        text-align: left;
        font-size: small;

        border-left: solid 4px #336;
        border-bottom: solid 1px #336
    }
    p             { line-height: 130% }    ――― p.66
❹ div.Main      {
        background: #d7d7d7;
        color: #330;
        margin: 0px 10% 10px 10%;
        padding: 10px;
        border-top: solid 2px #ccc;
        border-right: solid 2px #999;
        border-bottom: solid 2px #999;
        border-left: solid 2px #666
    }
```

```
div.Message   {
    margin: 0px 16% 10px 16%;
    padding: 10px
}
.photo          {
    margin-top: 5px;
    margin-left: 5px;
    float: right
}
.txtend         { clear: right }

/*=====テーブルへのスタイル設定=====*/
table           {
    background: #999;
    color: #000;
    padding: 1px;
    table-layout: fixed
}
td              {
    background: #369;
    color: #fff;
    padding: 5px;
    line-height: 130%
}
.param          {
    background: #336;
    color: #fff;
    text-align: center;
    border: inset 2px #336
}

/*=====フォームへのスタイル設定【注意1】=====*/
form            {
    border: inset 2px #336;
    margin: 0px;
    background: transparent;
    color: #000;
    padding: 5px;
    line-height: 150%
}
input.txt       {
    width: 200px;
    background: #369;
    color: #fff;
    ime-mode: active
}
input.button  {
    margin: 10px;
    background: #336;
    color: #fff
}
.color0         {
    background: #369;
    color: #fff
}
.color1         {
    background: transparent;
    color: #336
}
```

⑤ .photo
⑦ table / td / .param
⑧ input.txt
⑩ input.button
⑨ .color0 / .color1

float: right — p.216
clear: right — p.219
table-layout: fixed — p.254
ime-mode: active — p.323

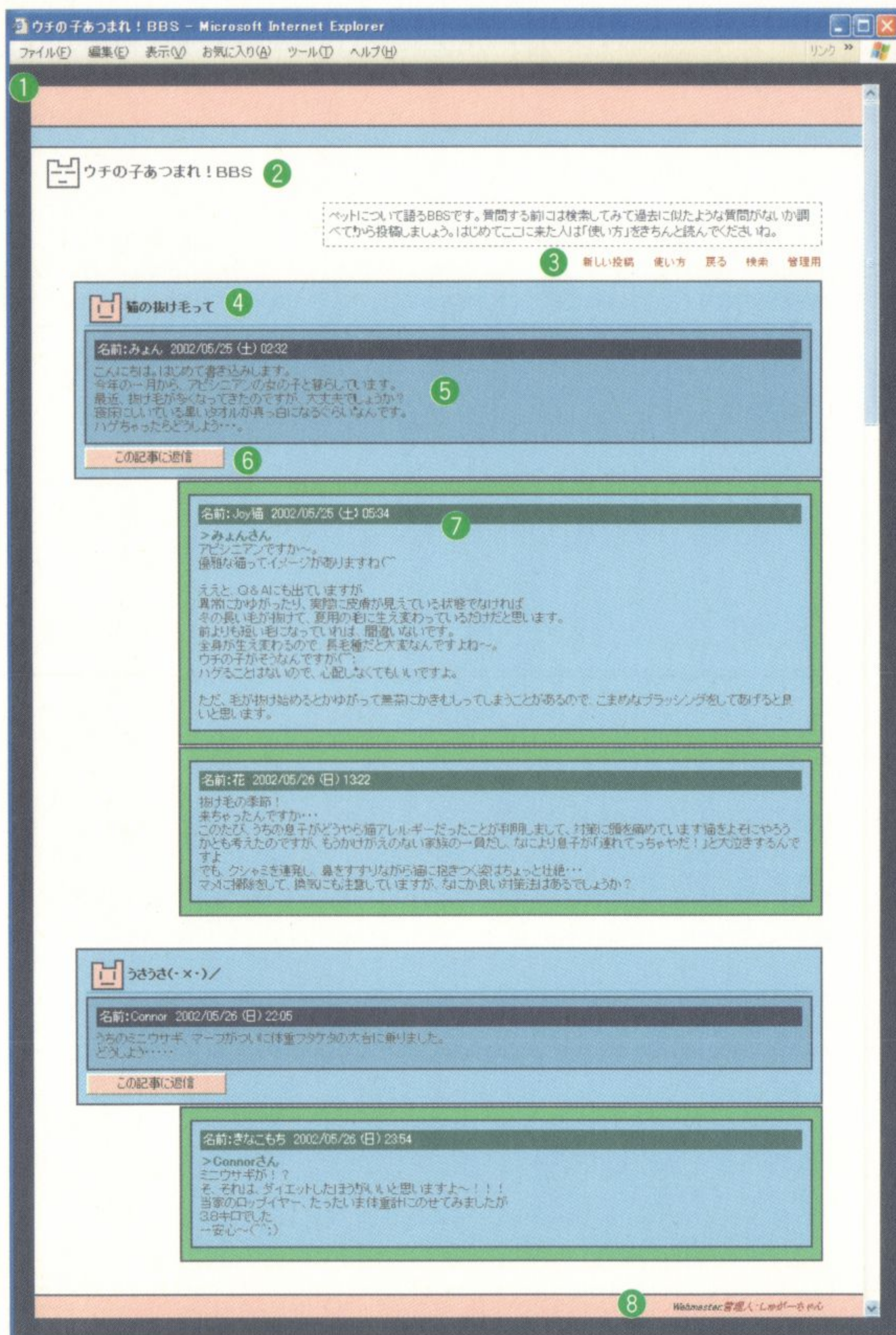
```
⑨ .color2          {
      background: transparent;
      color: #369
}
/*=====フォームへのスタイル設定ここまで=====*/

/* フッタの設定 */
div.Navigation{
      margin-top: 20px;
      margin-bottom: 20px;
      padding: 15px;
      padding-right: 10%;
      background: #003;
      color: #fff;
      border: #003 2px solid;
      text-align: right
}
div.Navigation a:link, div.Navigation a:visited {  /* div.Navigationの子となるa要素に適用 */ ── p.216
      background: transparent;
      color: #fff;
      text-decoration: none ── p.60
}
```

【注意1】このソースではフォームの要素へスタイルシートの設定を行っていますが、Netscape Navigator 4.x ではフォームの要素にスタイルシートを適用すると、入力エリアが表示されないなどのバグが発生します。これには、フォーム関係のスタイルのみを別のスタイルファイルに記述し、@import でそのスタイルファイルを読み込むという回避策があります。詳細は p.389 を参照してください。

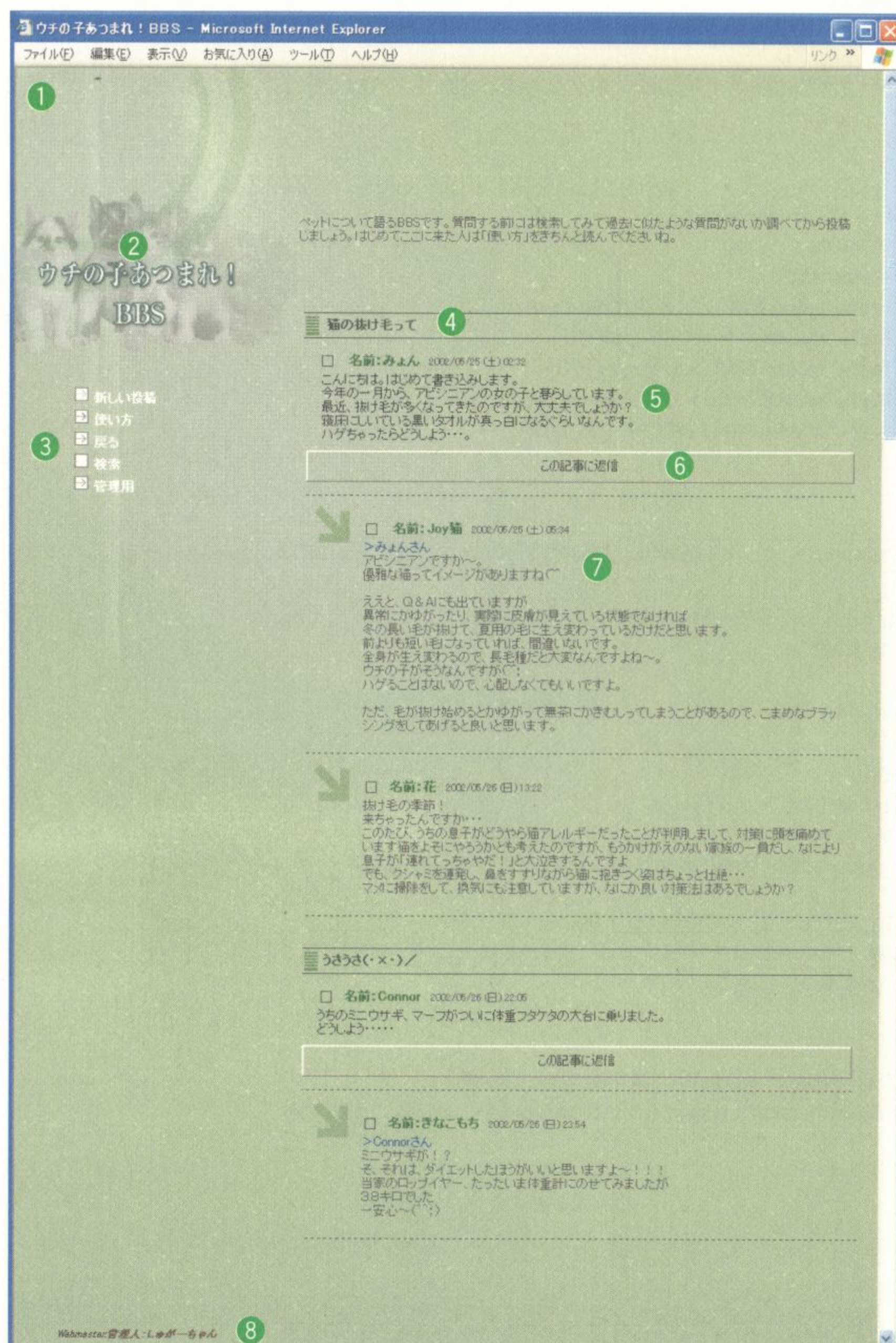
● 同一 HTML 文書に異なるスタイル適用する 1...掲示板

同じ HTML 文書に別の外部スタイルファイルを読み込んでまったく違う雰囲気のページを作ることもできます。以下のサンプルはそれぞれ外部スタイルファイルのみを変えてスタイル比較をしたものです。

最初に紹介するのは掲示板のページです。要素が繰り返される掲示板はテーブルによるレイアウトよりも、スタイルシートのほうがファイルサイズが軽くすむという利点があります。

「sample3.html」+「layout_a.css」の表示（サンプル A）

また、このサンプルではNetscape Navigator4.xで生じるバグを回避する方法の一例を使用しています。バグを招くスタイルシートをさらに別のスタイルファイルにまとめ、@importでそれを読み込むという方法です。NN4.xは@importに対応していないために問題のスタイルシートを読み込みまず、バグを回避することができます。

「sample3.html」+「layout_b.css」の表示(サンプルB)

サンプルA、Bに共通のHTML文書「sample3.html」

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv=Content-Style-Type content="text/css">
<link rel="stylesheet" href="layout_a.css">
<title>ウチの子あつまれ！ＢＢＳ</title>
</head>

❶ <body lang="ja">

<div id="head">
❷ <h1>ウチの子あつまれ！ＢＢＳ</h1>
<div class="contri">
ペットについて語るBBSです。質問する前には検索してみて……（中略）……くださいね。
</div>
</div>

<!-- ===== メニュー ===== -->
❸ <div id="menu">
<ul class="manulist">
  <li><a href="new.html">新しい投稿</a></li>
  <li><a href="use.html">使い方</a></li>
  <li><a href="return.html">戻る</a></li>
  <li><a href="search.html">検索</a></li>
  <li><a href="admin.html">管理用</a></li>
</ul>
</div>

<div id="content"><!-- === 記事全体に適用するスタイル（サンプルBのみ）=== -->

<!-- =====1スレッド開始===== -->
<div class="piece">

<div class="parents">                <!-- === 親記事ここから === -->

❹ <h2>猫の抜け毛って</h2>
❺ <div class="comimg">
<div class="name">
名前：みょん  <span class="date">2002/05/25 (土) 02:32</span>
</div>
こんにちは。はじめて書き込みします。<br>
……（中略）……
ハゲちゃったらどうしよう・・・。
</div>

❻ <form class="resbox" method="get" action="res.cgi">
  <input class="submit" type="submit" value="  この記事に返信  ">
</form>

</div>                <!-- === 親記事ここまで === -->

<hr>
```

サンプルBの場合、
<link rel="stylesheet" href="layout_b.css">
となります。

7

```
<div class="res">                    <!-- ===子記事ここから=== -->
<div class="rescomment">
<div class="resname">
名前：Joy猫　<span class="date">2002/05/25 (土) 05:34</span>
</div>
<span class="resmark">＞みょんさん</span><br>
アビシニアンですか～。<br>
……（中略）……
ブラッシングをしてあげると良いと思います。<br>
</div>
</div>                               <!-- ===子記事ここまで=== -->

<hr>

<div class="res">                    <!-- ===子記事ここから=== -->
<div class="rescomment">
<div class="resname">
名前：花　<span class="date">2002/05/26 (日) 13:22</span>
</div>
抜け毛の季節！<br>
……（中略）……
なにか良い対策法はあるでしょうか？<br>
</div>
</div>                               <!-- ===子記事ここまで=== -->

<div class="piecebottom"><hr></div>
</div>
<!-- =====1スレッドここまで===== -->


<!-- =====1スレッドここから===== -->
<div class="piece">

<div class="parents">                <!-- ===親記事ここから=== -->

<h2>うさうさ（・×・）／</h2>
<div class="comimg">
<div class="name">
名前：Connor　<span class="date">2002/05/26 (日) 22:05</span>
</div>
うちのミニウサギ、マーフがついに体重フタケタの大台に乗りました。<br>
どうしよう・・・・・</div>

<form class="resbox" method="get" action="res.cgi">
  <input class="submit" type="submit" value="　この記事に返信　">
</form>

</div>                               <!-- ===親記事ここまで=== -->

<hr>

<div class="res">                    <!-- ===子記事ここから=== -->
<div class="rescomment">
<div class="resname">
名前：きなこもち　<span class="date">2002/05/26 (日) 23:54</span>
</div>
<span class="resmark">＞Connorさん</span><br>
ミニウサギが！？<br>
……（中略）……
```

```
一安心～（＾＾；）<br>

</div>
</div>                              <!-- ===子記事ここまで=== -->

<div class="piecebottom"><hr></div>
</div>
<!-- =====1スレッドここまで===== -->

</div>

<!-- =====フッタ===== -->
<div id="footer">
<hr>
<address>
Webmaster:<a href="mailto:xxxx@xxxx.co.jp">管理人：しゅがーちゃん</a>
</address>
</div>

</body>
</html>
```

サンプルAのスタイルファイル「layout_a.css」

body要素にborderプロパティを設定し、ページ全体をボーダーで囲んだスタイルです。Windows版Internet Explorer5.5以上ではボーダーの内側にスクロールバーが表示されるので、一風変わったデザインとなります。

layout_a.css

```
@import url(sub_a.css)          /* 【注意1】 */

❶ body          {
    margin: 0;
    padding: 0;
    background: #fff;
    color: #000;
    border: 20px solid #336      /* 全体を紺色のボーダーで囲う */        p.195
}
a               {
    background: transparent;
    color: #933;
    text-decoration: none
}
a:active, a:hover{
    background: #9cf;
    color: #336;
    text-decoration: underline
}
hr              { display: none }   /* 【注意2】 */                     p.202
#head           {
    margin: 0;
    padding: 55px 0 0 0;                                               p.170
    background: url("headbg.gif") #fff repeat-x;
    color: #000
}
❷ h1            {
    margin: 10px;
    padding: 11px 40px;
    font-size: 14px;
    background: url("title_a.gif") #fff no-repeat;
    color: #336
}
❹ h2            {
    margin: 0;
    padding: 9px 40px;
    font-size: 12px;
    background: url("h2.gif") #9cf no-repeat;
    color: #000;
    border-bottom: 2px solid #76a9dc
}
div.contri {
    margin: 10px 5% 10px 35%;                                          p.164
    padding: 5px;
    background: #fff;
    color: #336;
    font-size: small;
    line-height: 120%;
    border: 1px dotted #336
}
```

```
/*=====メニューに関するスタイル設定=====*/
❸ #menu         {
        padding: 0;
        margin: 0 5%;
        font-size: 90%;
        background: transparent;
        color: #999;
        text-align: right
   }
❸ #menu ul      { /*  【注意3】  */ }
❸ #menu ul li   { /*  【注意3】  */ }

/*=====記事に関するスタイル設定=====*/
div.piece       { margin: 10px 5% }
      /*=====親記事=====*/
div.parents     {
        margin: 3px 0 1px 0;
        padding: 8px;
        background: #9cf;
        color: #333;
        border: 2px solid #336
   }
❺ div.comimg    {
        margin: 3px 0;
        padding: 8px;
        background: #84b0dc;
        color: #333;
        border: 2px solid #336
   }
❺ div.name      {
        margin: 0 0 5px 0;
        padding: 3px;
        background: #336;
        color: #fff;
        border: 0;
        font-size: small
   }
❻ form.resbox   {
        margin: 0;
        padding: 0
   }
❻ input.submit  {
        font-size: 12px;
        background: #fcc;
        color: #000
   }
      /*=====子記事=====*/
❼ div.res       {
        margin: 1px 0 1px 100px;
        padding: 8px;
        background: #6c9;
        color: #333;
        border: 2px solid #336
   }
❼ div.rescomment {
        margin: 3px 0;
        padding: 8px;
        background: #9cf;
        color: #333;
```

p.117

```
        border: 2px solid #336
    }
❼  div.resname    {
        margin: 0 0 5px 0;
        padding: 3px;
        background: #366;
        color: #fff;
        border: 0;
        font-size: small
    }
❼  .resmark       {
        background: transparent;
        color: #366;
        font-weight: bold
    }
    div.piecebottom { margin: 30px }

❽  #footer        {                /* フッタ */
        font-size: 10px;
        margin: 5px 0 0 0;
        padding: 5px 5%;
        background: #fcc;
        color: #000;
        border-top: 2px solid #336;
        text-align: right
    }
```

font-weight ― p.122

【注意1】@importで別のスタイルファイルを読み込むには@import:url(★)という形式で読み込むスタイルファイルを記述します。@importの指定は、ほかのスタイルの設定よりも前に置かなければ有効になりません。

【注意2】<hr>タグはスタイルシートが無効の環境やブラウザでの位置調整のために設定していますので、スタイルシートが有効の場合は表示しないように設定しています。

【注意3】ここではHTML文書のリスト要素（ul、li要素）を1行で表示したいのですが、ここにdisplay:inlineを指定するとNetscape Navigator 4.xで要素が重なり合ってしまうというバグが発生するため、別のスタイルファイル「sub_a.css」に指定します。

バグ回避のためのスタイルファイル「sub_a.css」

Netscape Navigator4.xでバグが発生するスタイルシートを別のスタイルファイルにまとめ、「layout_a.css」の@importによって読み込みます（上記【注意】参照）。

sub_a.css

```
❸  #menu ul       {
        display: inline;
        list-style-type: none;
        padding: 0 0 0 5px;
        margin: 0
    }
❸  #menu ul li    {
        display: inline;
        list-style-type: none;
        padding: 0 0 0 5px;
        margin: 0 0 0 10px
    }
```

display ― p.202
list-style-type ― p.242

サンプルBのスタイルファイル「layout_b.css」

メニュー部分にfloatプロパティを指定し、その他の要素を回り込ませて段組のようにレイアウトしたスタイルです。このような段組のレイアウトはposition:absoluteを利用しても実現可能ですが、floatなら文字が増えてもHTMLを修正するだけで済みます（positionの場合はスタイルシートを変更する必要があります）。そのため掲示板のような文字が増えていくコンテンツにはfloatが向いていると言えるでしょう。ただしブラウザによるバグが多いので注意が必要です。

layout_b.css

```
@import url(sub_b.css)          /* 【注意1】 */

❶ body          {
     margin: 0px;
     padding: 0;
     background: url("bg_g.jpg") #b3c2a3 no-repeat;
     color: #000
}
a               {
     background: transparent;
     color: #630;
     text-decoration: none;
     font-weight: bold
}
a:active, a:hover{
     background: transparent;
     color: #fff;
     text-decoration: none;
     font-weight: bold
}
hr              { display: none }  /* 【注意2】 */
#head           {                  /* レイアウト指定 */
     margin: 80px 0 10px 30%;                                  ──p.164
     padding: 55px 20px 20px 20px                              ──p.170
}
❷ h1            { display: none }  /* 【注意3】 */
❹ h2            {                  /* 【注意4】 */
     margin: 10px 0 5px 0;
     padding: 5px 20px;
     font-size: 12px;
     border-top: 1px solid #030;
     border-bottom: 1px solid #030
}
div.contri      {
     margin: 0;
     padding: 5px;
     background: transparent;
     color: #333;
     font-size: 12px
}

/*=====メニューに関するスタイル設定=====*/
❸ #menu         {                  /* レイアウト指定 */
     float: left;                                              ──p.216
     width: 30%;
     margin-top: 0;
     padding: 80px 0 0 20px
```

```
}
❸ #menu ul {
      display: block;                                   p.202
      margin: 10px 15px;
      padding: 20px ;
      list-style: inside url("arrow.gif");              p.250
      font-size: 12px;
      line-height: 180%;
      background: transparent;
      color: #fff
      }
❸ #menu a {
      display: inline;
      margin: 3px;
      padding: 3px;
      background: transparent;
      color: #fff
}
❸ #menu  a:hover {
      display: inline;
      margin: 3px;
      padding: 3px;
      background: #b3c2a3;
      color: #030
}

/*=====記事に関するスタイル設定=====*/
#content        {                        /* 記事全体のレイアウト指定 */
      margin-left: 31%;                                 p.160
      padding: 20px
}
       /*=====親記事=====*/
div.parents     {
      margin: 0 5px 0 0;
      border-bottom: 1px dotted #030                    p.192
}
❺ div.comimg    {
      margin: 3px 0;
      padding: 5px 0 5px 15px
}
❺❼div.name, div.resname          {     /* 【注意4】 */
      margin: 0;
      padding: 5px 0 5px 26px;
      border: 0;
      font-size: 12px;
      font-weight: bold
}
❺❼span.date {
      background: transparent;
      color: #333;
      font-size: 10px;
      font-weight: normal
}
❻ form.resbox    {
      margin: 2px 0 12px 0;
      padding: 0
}
❻ input.submit  {
      font-size: 12px;
```

```
        background: transparent;
        color: #000;
        width: 100%;
        margin: 0;
        padding: 5px
}
        /*===== 子記事 =====*/
⑦ div.res          {                    /* 【注意4】 */
        margin: 1px 0;
        padding: 8px 8px 8px 50px;
        border-bottom: 1px dotted #030
}
⑦ div.rescomment {
        margin: 3px 0;
        padding: 8px
}
⑦ .resmark{
        background: transparent;
        color: #039
}
div.piecebottom   { margin: 30px }

⑧ #footer{                              /* フッタ */
        clear: left;                    /* レイアウト指定（#menuのfloatを解除） */   p.219
        font-size: 10px;
        margin: 30px 0 0 0;
        padding: 5px 5%;
        background: transparent;
        color: #000
}
```

【注意1】【注意2】p.395の注意を参照してください。

【注意3】h1要素の内容はbody要素の背景画像「bg_g.jpg」に画像として含まれますので、非表示にします。

【注意4】この3カ所には背景画像の指定も行いますが、Netscape Navigator 4.xでは、背景画像に透過GIFを指定しても正常に透過しないというバグが発生します。これを回避するために、背景画像のスタイルに関してはこのスタイルファイルでは指定せず、別のスタイルファイル「sub_b.css」に指定します。

バグ回避のためのスタイルファイル「sub_b.css」

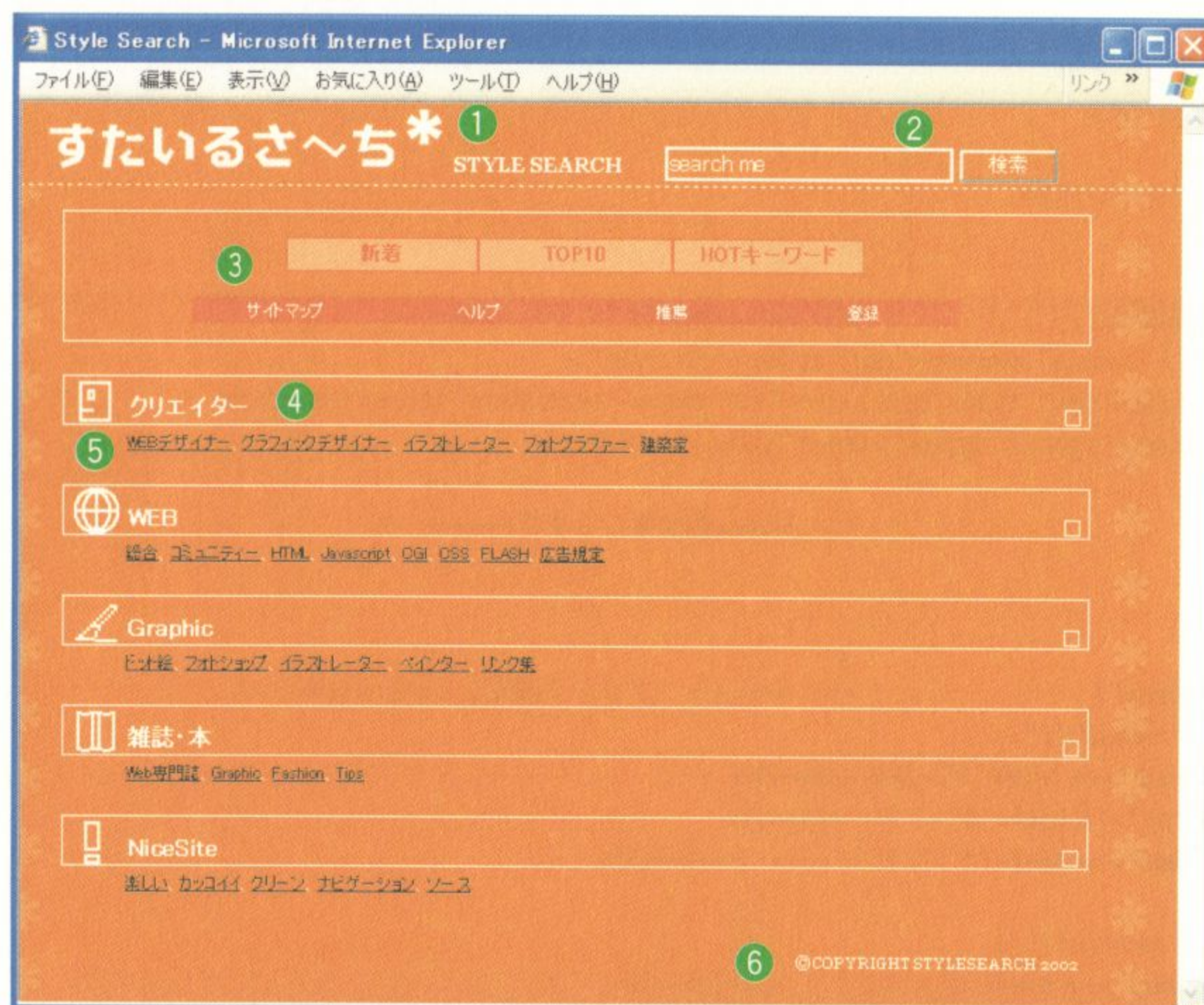
Netscape Navigator4.xでバグが発生するスタイルシートを別のスタイルファイルにまとめ、「layout_b.css」の@importによって読み込みます（上記【注意】参照）。

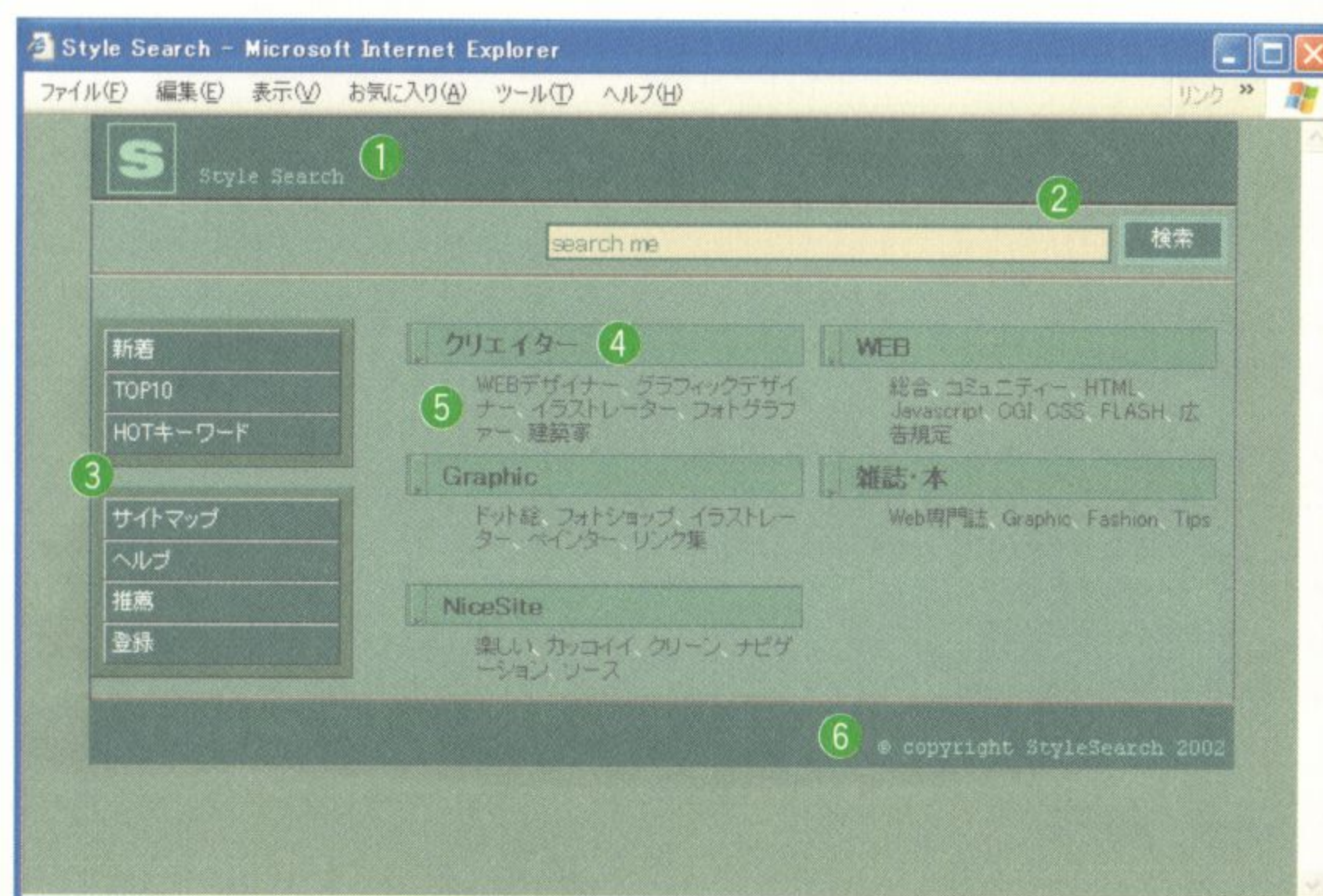
sub_b.css

```
④ h2            {
        background: url("h2_b.gif") no-repeat;
        color: #030
}
⑤⑦div.name,div.resname {
        background: url("name.gif") no-repeat left center ;
        color : #063
}
⑦ div.res {
        background: url("res.gif") no-repeat;          p.157
        color : #333
}
```

● 同一HTML文書に異なるスタイル適用する2...ポータルサイト

ポータルサイトのサンプルです。トップリンク部分はHTML文書ではリストとして記述されていますが、スタイルシートによって横並び、縦並びというようにまったく違うスタイルで表示することが可能です。

「sample4.html」+「layout_c.css」の表示（サンプルC）

「sample4.html」+「layout_d.css」の表示（サンプルD）

サンプルC、Dに共通のHTML文書「sample4.html」

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv=Content-Style-Type content="text/css">
<link rel="stylesheet" href="layout_c.css">
<title>Style Search</title>
</head>

<body lang="ja">

<h1>Style Search</h1>
<hr>

<div id="toolbox">                <!-- =====検索窓ここから===== -->
<form class="search" method="get" action="/search/">
  <input class="text" type="text" size="20" name="searchbox" value="search me">
  <input class="submit" type="submit" value="  検索  ">
</form>
</div>                            <!-- =====検索窓ここまで===== -->

<hr>

<div id="menu">                   <!-- =====トップリンク部分ここから===== -->
<ul class="menulist1">
  <li id="new"><a href="#">新着</a></li>
  <li id="top"><a href="#">TOP10</a></li>
  <li id="hot"><a href="#">HOTキーワード</a></li>
</ul>
<ul class="menulist2">
  <li id="map"><a href="#">サイトマップ</a></li>
  <li id="help"><a href="#">ヘルプ</a></li>
  <li id="suisen"><a href="#">推薦</a></li>
  <li id="touroku"><a href="#">登録</a></li>
</ul>
</div>                            <!-- =====トップリンク部分ここまで===== -->

<hr>

<!-- =====メイン部分ここから===== -->
<div id="content">

<div class="content1">
<div class="content-left">
  <dl>
  <dt id="cri"><a href="#">クリエイター</a></dt>
    <dd><a href="#">WEBデザイナー</a>、
    <a href="#">グラフィックデザイナー</a>、
    <a href="#">イラストレーター</a>、
    <a href="#">フォトグラファー</a>、
    <a href="#">建築家</a></dd>
  </dl>
</div>
<div class="content-right">
  <dl>
```

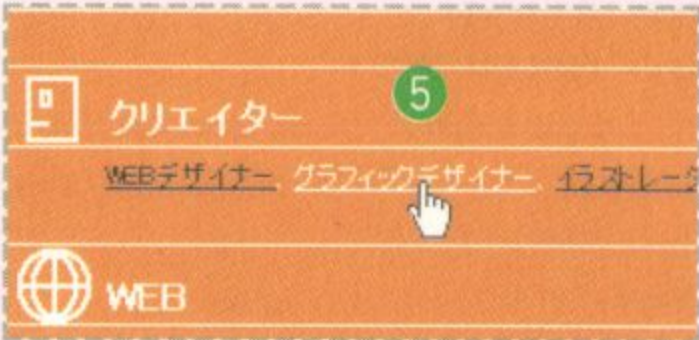
サンプルDの場合、
<link rel="stylesheet" href="layout_d.css">
となります。

```
  <dt id="web"><a href="#">WEB</a></dt>
    <dd><a href="#">総合</a>、
    <a href="#">コミュニティー</a>、
    <a href="#">HTML</a>、
    <a href="#">Javascript</a>、
    <a href="#">CGI</a>、
    <a href="#">CSS</a>、
    <a href="#">FLASH</a>、
    <a href="#">広告規定</a></dd>
  </dl>
</div>
</div>

<div class="content2">
<div class="content-left">
  <dl>
  <dt id="gra"><a href="#">Graphic</a></dt>
    <dd><a href="#">ドット絵</a>、
    <a href="#">フォトショップ</a>、
    <a href="#">イラストレーター</a>、
    <a href="#">ペインター</a>、
    <a href="#">リンク集</a></dd>
  </dl>
</div>
<div class="content-right">
  <dl>
    <dt id="book"><a href="#">雑誌・本</a></dt>
    <dd><a href="#">Web専門誌</a>、
    <a href="#">Graphic</a>、
    <a href="#">Fashion</a>、
    <a href="#">Tips</a></dd>
  </dl>
</div>
</div>

<div class="content3">
<div class="content-left">
  <dl>
  <dt id="nice"><a href="#">NiceSite</a></dt>
    <dd><a href="#">楽しい</a>、
    <a href="#">カッコイイ</a>、
    <a href="#">クリーン</a>、
    <a href="#">ナビゲーション</a>、
    <a href="#">ソース</a></dd>
  </dl>
</div>
</div>

</div>
<!-- =====メイン部分ここまで===== -->

<hr>

⑥ <div class="copy" lang="en">&copy; copyright StyleSearch 2002</div>

</body>

</html>
```

サンプルCのスタイルファイル「layout_c.css」

横幅を固定した（ウィンドウの大きさを変更してもレイアウトは変わらない）デザインのポータルサイトのサンプルです。

トップリンクのli要素にdisplay:inlineを指定して上部に横に並ぶようにしています。

なお、このサンプルはNetscape Navigator4.xでは表示が崩れる等のバグが発生します。

layout_c.css

```
body            {
    margin: 0px;
    padding: 0px;
    background: url("searchbg.gif") #f60 repeat-y;
    color: #fff
}
hr              { display: none }   /*  【注意1】  */
❶ h1            {
    margin: 0px 0px 15px 0px;
    border-bottom: 1px dotted #fff;
    background: url("title_a.gif") transparent no-repeat left center;
    color:#fff;
    font-size: 12px;
    font-family: Georgia, "Times New Roman", Times, serif;
    text-transform: uppercase;
    padding: 30px 5px 5px 270px
}

/*=====リンクのスタイル設定=====*/
❺ a             {
    background: transparent;
    color: #000;
    text-decoration: underline
}
❺ a:active, a:hover  {                                    p.39
    background: transparent;
    color: #fff;
    text-decoration: underline
}

/*=====検索窓のスタイル設定=====*/
❷ #toolbox      {
    margin: 20px 0px;
    padding: 0px;
    position: absolute;                                    p.208
    top: 5px;                                              p.212
    left: 400px;                                           p.212
    width: 300px;
    background: transparent;
    color: #fff;
    overflow: visible;                                     p.230
    z-index: 1                                             p.222
}
❷ input.text    {
    width: 180px;
    background: #f60;
    color: #fff;
    font-size: 14px;
```

```
        border: 2px solid #fff
    }
② input.submit  {
        width: 60px;
        font-size: 12px;
        background: #f60;
        color: #fff
    }

    /*=====トップリンク部分（上部の囲み）===== */
③ #menu          {
        margin: 10px 25px;
        padding: 0px;
        border: 1px solid #fff;
        width: 640px;
        background: #f60;
        color: #f60
    }
③ ul.menulist1  {
        margin: 15px 0px;
        padding: 0px;
        font-size: 12px;
        line-height: 150%;
        text-align: center;
        position: relative
    }
③ ul.menulist2  {
        margin: 10px 0px;
        padding: 0px;
        font-size: 10px;
        line-height: 150%;
        text-align: center;
        position: relative
    }
③ li             {
        width: 120px;
        display: inline;
        margin: 0px;
        padding: 0px
    }
        /*=====トップリンク内のリンクのスタイル設定====== */
③ .menulist1 li a{
        width: 100%;
        text-decoration: none;
        border: 1px solid #f30;
        font-weight: bold;
        background: #ffb27f;
        color: #f30
    }
③ .menulist2 li a{
        width: 100%;
        text-decoration: none;
        border: 1px dotted #f30;
        background: #f30;
        color: #fff
    }
③ .menulist1 a:active, .menulist1 a:hover,.menulist2 a:active, .menulist2 a:hover   {
        text-decoration: underline;
        background:  #fff;
```

p.198

p.202

```
    color: #f30
}

/*===== メイン部分の項目 =====*/
dl              {
    margin: 15px 25px;
    padding: 0px;
    width: 640px;
    background: transparent;
    color: #fff
}
④ dt            {
    margin: 5px 0px;
    padding: 12px 5px 2px 40px;
    border: 1px solid #fff
}
④ dt a          {
    text-decoration : none;
    font-weight : bold;
    font-size: 14px;
    background: transparent;
    color:#fff
}
⑤ dd            { font-size: 10px }
    /*===== 項目のタイトル =====*/
④ #cri          {
    background: url("ct_cri.gif") #f60 no-repeat;
    color: #fff
}
#web            {
    background: url("ct_web.gif") #f60 no-repeat;
    color: #fff
}
#gra            {
    background: url("ct_gra.gif") #f60 no-repeat;
    color: #fff
}
#book           {
    background: url("ct_book.gif") #f60 no-repeat;
    color: #fff
}
#nice           {
    background: url("ct_nice.gif") #f60 no-repeat;
    color:#fff
}


⑥ div.copy      {
    margin: 30px 25px;
    padding: 5px;
    font-family: Georgia, "Times New Roman", Times, serif;
    text-transform: uppercase;
    font-size: 10px;
    width: 640px;
    text-align: right
}
```

p.63

【注意 1】p.395 の注意 2 を参照してください。

サンプルDのスタイルファイル「layout_d.css」

横幅を変更可能な（ウィンドウの大きさを変更するとレイアウトも変わる）デザインのポータルサイトのサンプルです。

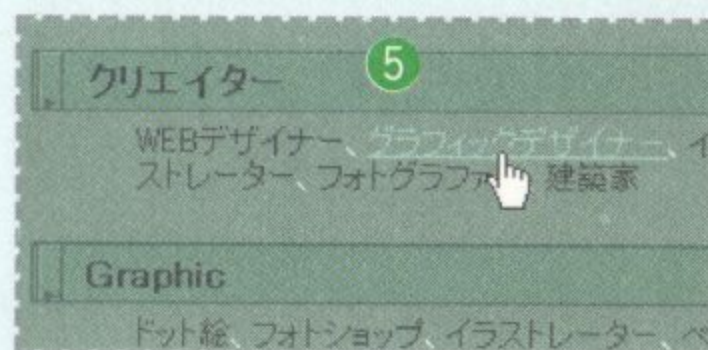
サンプルBと同じく段組のレイアウトになっていますが、こちらはposition:absoluteで制御しています。この場合、コンテンツの量が増えるなどして高さが変わるとスタイルシートを書き直す必要がありますが、コンテンツが頻繁に増えるのでなければ、position:absoluteでも問題はないでしょう。

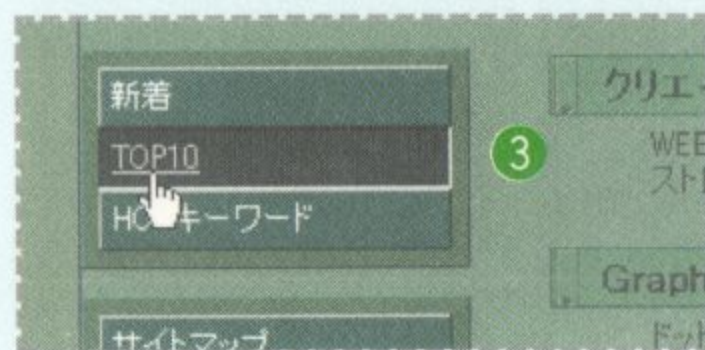
なお、このサンプルはNetscape Navigator4.xでは表示が崩れる等のバグが発生します。

SOURCE

```
body        {
    margin: 0px;
    padding: 0px;
    background: #7d9588;
    color: #ccc
}
hr          { display: none }   /* 【注意1】 */
❶ h1        {
    margin: 0px  5% 0px 5%;
    padding: 24px 5px 5px 60px;
    font-size: 12px;
    font-family: "Courier New", Courier, mono;
    font-weight: normal;
    border: 1px solid #336;
    border-bottom: 2px ridge #333;
    background: url("title_b.gif") #366 no-repeat left bottom;
    color: #9cc
}

/*=====リンクのスタイル設定=====*/
❺ a         {
    background: transparent;
    color: #033;
    text-decoration: none
}
❺ a:active, a:hover  {
    background: transparent;
    color: #9cc;
    text-decoration: underline
}

/*=====検索窓のスタイル設定=====*/
❷ form      {
    margin: 0px;
    padding: 0px
}
❷ #toolbox  {
    margin: 0px  5% 0px 5%;
    padding: 5px 5px 5px 20px;
    border-top: 1px solid #D0D0B9;
    border-bottom: 3px ridge #D0D0B9;
    text-align: right;
    background: #7a978d;
    color: #fff;
```

p.114

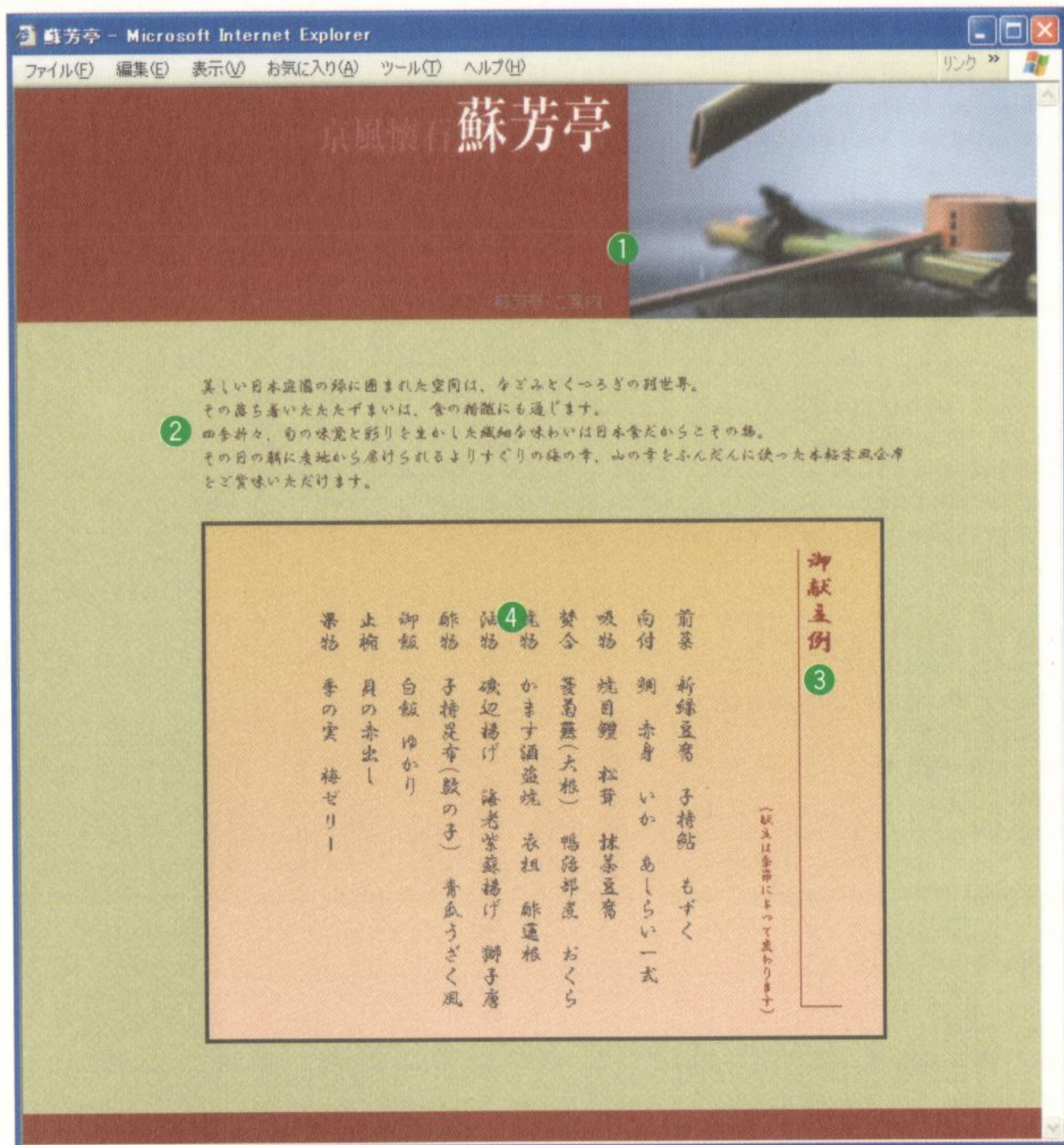
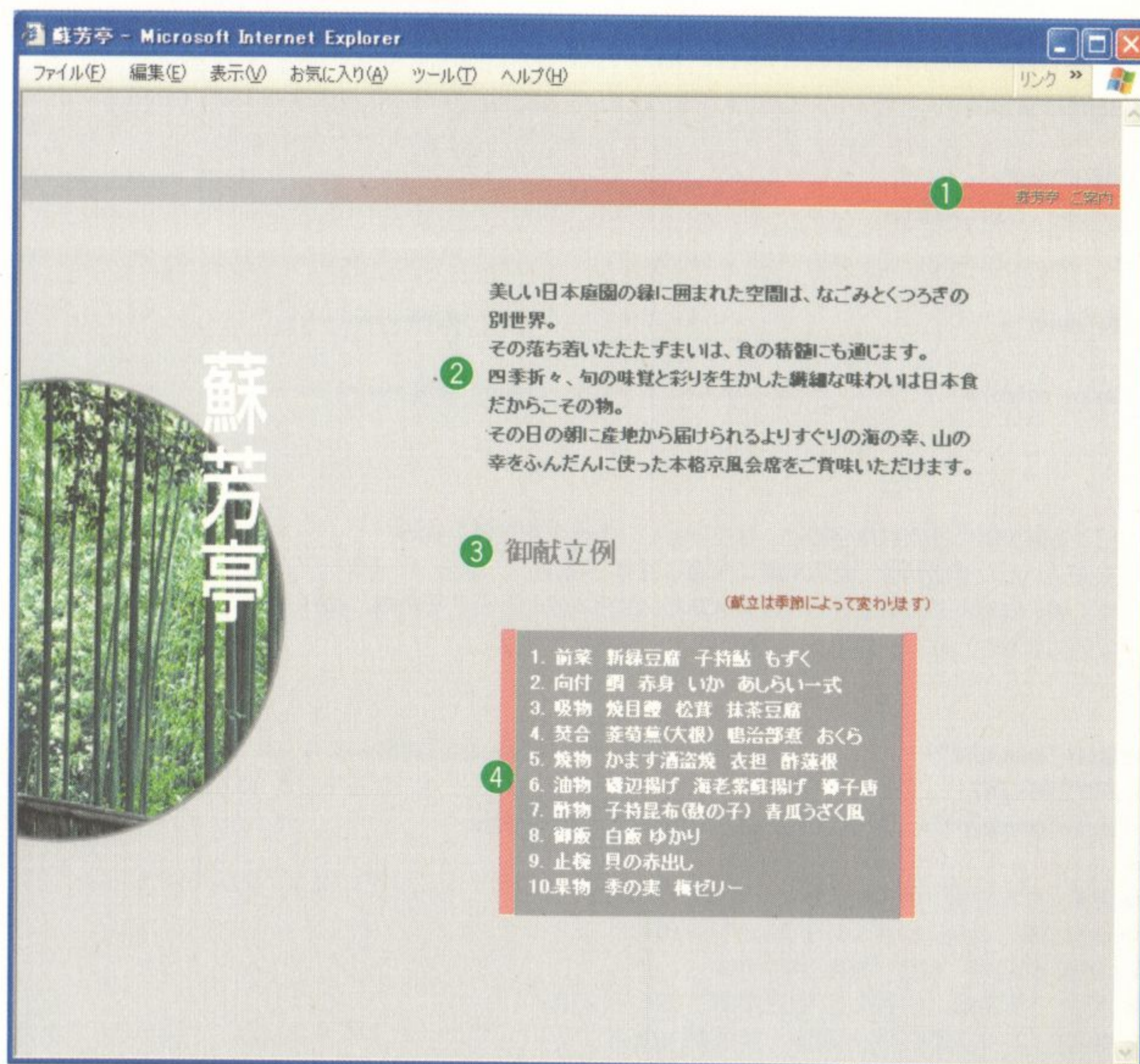
p.60

```
    border: 1px solid #336
}
❷ input.text    {
    margin: 0px;
    padding: 2px;
    width: 320px;
    font-size: 14px;
    background: #eaddc3;
    color: #366;
    border: 2px solid #366
}
❷ input.submit  {
    padding: 2px;
    width: 60px;
    font-size: 12px;
    background: #366;
    color: #fff;
    border: 3px double #9cc
}

/*=====トップリンク部分（左側）===== */
❸ #menu         {
    position: absolute;      ——— p.208
    z-index: 2;
    top:100px;               ——— p.212
    left: 5%;                ——— p.212
    width: 150px;
    margin: 1px;
    padding: 0px
}
❸ #menu ul li   { display: inline }
❸ #menu ul      {
    margin: 10px 0px;
    padding: 0.5em;
    list-style-type: none;   ——— p.242
    border-top: solid 1px #699;
    border-right: solid 1px #033;
    border-bottom: solid 1px #033;
    border-left: solid 1px #699;
    font-size: 12px;
    background: #547768;
    color: #333
}
    /*=====トップリンク内のリンクのスタイル設定====== */
❸ #menu a:link, #menu a:visited  {   ——— p.39
    margin: 0px;
    padding: 4px;
    border-top: solid 1px #ccc;
    border-right: solid 1px #033;
    border-bottom: solid 1px #033;
    border-left: solid 1px #ccc;
    display: block;
    background: #366;
    color: #fff
}
❸ #menu a:hover  {
    margin: 0px;
    padding: 4px;
    border-top: solid 1px #111;
    border-right: solid 1px #eee;
```

新着
TOP10
HOTキーワード
サイトマップ
クリエ
Graph
❸

```
        border-bottom: solid 1px #eee;
        border-left: solid 1px #111;
        display: block;
        background: #033;
        color: #ccc
    }

    /*=====メイン部分のスタイル設定=====*/
❹❺#content        {
        margin: 0px  5% 0px 5%;
        padding: 15px 0px 0px 180px;
        position: relative;                                   p.208
        z-index: 1;                                           p.222
        border: 1px solid #336;
        border-top: 1px solid #d0d0B9;
        border-bottom: 3px ridge #d0d0B9;
        text-align: left;
        background: #7a978d;
        color: #fff
    }
❹❺#content dl     { margin-top: 10px }
❹  #content dt     {                    /*  項目のタイトル  */
        margin: 5px 0px;
        padding: 4px 5px 2px 20px;
        border: 1px solid #366;
        font-size: 14px;
        font-weight: bold;
        background: url("dt_b.gif") #6d9588 no-repeat;
        color: #fff
    }
❺  #content dd     { font-size: 12px }
        /*=====項目のレイアウト=====*/
❹❺div.content1,div.content2,div.content3   {
        display: block;
        position: relative;
        width: 100%;
        height: 6em
    }
❹❺.content-left   {
        position: absolute;
        width: 48%
    }
    .content-right {
        position: absolute;
        left: 50%;
        width: 48%
    }

❻  div.copy        {
        margin: 0px  5% 0px 5%;
        padding: 15px 5px 5px 30px;
        font-size: 12px;
        font-family: "Courier New", Courier, mono;
        text-align: right;
        background: #366;
        color: #9cc
    }
```

【注意1】p.395の注意2を参照してください。

● 同一HTML文書に異なるスタイル適用する3...おしながき

テキストを縦書きに配置した場合と横書きに配置した場合のサンプルです。スタイルに合わせてフォントを変えるだけで、異なった雰囲気を演出できます。ただし、縦書き表示できるのはInternet Explorer 5.5以上のみです。

「sample5.html」＋「layout_e.css」の表示（サンプルE）

「sample5.html」＋「layout_f.css」の表示（サンプルF）

サンプルE、Fに共通のHTML文書「sample5.html」

SOURCE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="ja">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
<meta http-equiv=Content-Style-Type content="text/css">
<link rel="stylesheet" href="layout_e.css">
<title>蘇芳亭</title>
</head>

<body lang="ja">

❶ <div class="top">
<h1>蘇芳亭　ご案内</h1>
</div>

<div id="main">                    <!-- =====スタイル「main」適用===== -->

<div class="catch">                <!-- =====スタイル「catch」適用===== -->
<hr>

❷ <p>
美しい日本庭園の緑に囲まれた空間は、なごみとくつろぎの別世界。<br>
その落ち着いたたたずまいは、食の精髄にも通じます。<br>
四季折々、旬の味覚と彩りを生かした繊細な味わいは日本食だからこその物。<br>
その日の朝に産地から届けられる……（中略）……いただけます。
</p>

<div class="menulist">             <!-- =====スタイル「menulist」適用===== -->
❸ <h2>御献立例</h2>
<div class="comment">（献立は季節によって変わります）</div>
❹ <ol >
  <li>前菜　新緑豆腐　子持鮎　もずく</li>
  <li>向付　鯛　赤身　いか　あしらい一式</li>
  <li>吸物　焼目鱧　松茸　抹茶豆腐</li>
  <li>焚合　菱菊蕪（大根）　鴨治部煮　おくら</li>
  <li>焼物　かます酒盗焼　衣担　酢蓮根</li>
  <li>油物　磯辺揚げ　海老紫蘇揚げ　獅子唐</li>
  <li>酢物　子持昆布（数の子）　青瓜うざく風</li>
  <li>御飯　白飯 ゆかり</li>
  <li>止椀　貝の赤出し</li>
  <li>果物　季の実　梅ゼリー</li>
</ol>
</div>                             <!-- =====スタイル「menulist」適用ここまで===== -->

</div>                             <!-- =====スタイル「catch」適用ここまで===== -->

</div>                             <!-- =====スタイル「main」適用ここまで===== -->

<hr>

<div class="foot"> </div>

</body>
</html>
```

サンプルFの場合、
<link rel="stylesheet" href="layout_f.css">
となります。

サンプルEのスタイルファイル「layout_e.css」

縦書き表示と行書体、明朝体（サンセリフ系）のフォント指定をして、コンテンツに合わせたデザインを実現しています。ただし、縦書きは対応ブラウザがInternet Explorer5.5以上、フォントの指定はユーザーの環境に左右される、など条件が多くなります。そのあたりも考慮に入れてスタイルシートを作成するようにしましょう。

layout_e.css

```
body           {
     margin: 0px;
     padding: 0px;
     background: url("bg_a.gif") #cc9 repeat;
     color: #000
}
hr             {
     display: none;               /* 【注意1】 */
     margin: 5px
}

❶ div.top      {
     padding: 0px 300px 0px 0px;
     background: url("image.jpg") #000 no-repeat right;
     color: #fff;
     text-align: right
}
❶ h1           {
     margin: 0px;
     padding: 150px 20px 5px 5px;
     background: url("title_a.gif") #900 no-repeat top right;   ―― p.157
     color: #966;
     text-align: right;                                         ―― p.69
     font-weight: lighter;                                      ―― p.122
     font-size: 12px
}

❷❸❹#main       {
     margin: 0px 10% 30px 15%;
     padding: 20px;
     background: transparent;
     color: #630;
     line-height: 140%
}

❷❸❹div.catch   {                 /* 【注意2】 */
     margin: 0px;
     padding: 0px;
     background: transparent;
     color: #630;
     text-align: left;
     font-family: "ＤＦ行書体","ＭＳ Ｐ明朝", "細明朝体", serif   ―― p.114
}

❸❹div.menulist {                 /* 【注意2】 */
     width: 500px;
     writing-mode: tb-rl;                                       ―― p.100
     font: 16px/180% "ＤＦ行書体","ＭＳ Ｐ明朝", "細明朝体",Osaka,serif;
```

```
        text-align: left;
        background: url("bg_a2.gif") #cc9 repeat;
        color: #633c3c;
        padding: 10px 30px 10px 10px;
        border: #630 3px ridge
    }
❸  h2              {                    /*  【注意2】  */
        margin: 10px;
        background: transparent;
        color: #900;
        text-align: left;
        font-size: 18px;
        font-family: "ＤＦ行書体","ＭＳ Ｐ明朝", "細明朝体", serif;
        border-left: #900 1px solid;
        border-bottom: #900 1px solid
    }
    div.comment {                       /*  【注意2】  */
        background: transparent;
        color: #900;
        margin: 0px;
        text-align: right;
        font-size: 10px;
        font-family: "ＤＦ行書体","ＭＳ Ｐ明朝", "細明朝体", serif
    }

❹  ol              { padding: 10px }
❹  li              { list-style-type: none }

    div.foot        {
        background: #900;
        color: #fff;
        margin: 0px;
        padding: 10px;
        border: #900 1px solid
    }
```

p.56

【注意1】p.395の注意を参照2してください。

【注意2】Netscape Navigator4.xでは指定内に日本語（フォント名）が含まれるこれらのスタイルは反映されません。

サンプルFのスタイルファイル「layout_f.css」

横書きのサンプルです。ゴシック系（セリフ系）のフォントを指定しています。

layout_f.css

```
body          {
      margin: 0px;
      padding: 0px;
      background: #ddd;
      color: #000
}
hr            {
      display: none;                /* 【注意1】 */
      margin: 5px
}

❶ div.top     {
      margin: 0px;
      padding: 0px;
      background: #ddd;
      color: #fff;
      border: 1px
}
❶ h1          {
      margin: 56px 0px 0px 0px;     /* 【注意2】 */  ———— p.164
      padding: 3px 5px;
      background: url("bg_b.jpg") #f66 repeat-y;
      color: #666;
      text-align: right;
      font-weight: lighter;
      font-size: 10px
}

❷❸❹#main      {
      margin: 0px;
      padding: 10px 10% 30px 300px;   /* 【注意2】 */  ———— p.170
      background: url("mainbg.gif") #ddd no-repeat left center;
      color: #030;
      font-size: 12px;
      line-height: 150%
}

❷❸❹div.catch  {                     /* 【注意3】 */
      margin: 0px;
      padding: 20px;
      background: transparent;
      color: #030;
      text-align: left;
      font-family: "ＭＳ Ｐゴシック",Osaka,"MS Gothic", serif;  ———— p.114
      font-weight: bold
}

❸❹div.menulist  { padding: 10px 30px 10px 0px }
❸ h2          {                     /* 【注意3】 */
      margin: 12px;
      background: transparent;
      color: #666;
      text-align: left;
```

```
    font-size: 18px;
    font-family: "ＭＳ Ｐゴシック",Osaka,"MS Gothic", serif
}
div.comment  {                    /* 【注意3】 */
    margin: 0px;
    background: transparent;
    color: #900;
    text-align: right;
    font-size: 10px;
    font-family: "ＭＳ Ｐゴシック",Osaka,"MS Gothic", serif;
    font-weight: normal
}

ol           {
    margin: 10px;
    padding: 10px;
    background: #999;
    color: #fff;
    font-size: 12px;
    line-height: 140%;
    border-left: #f99 10px solid;
    border-right: #f99 10px solid
}
li           { list-style-position: inside }

div.foot     { padding: 10px }
```

④ ol, ④ li

p.248

【注意1】p.395の注意2を参照してください。

【注意2】Netscape Navigator4.xではmarginやpaddingを個別に指定すると反映されないことがあるので、まとめて指定したほうが安全です。

【注意3】p.412の注意2を参照してください。

5 APPENDIX

適用・デフォルト・継承一覧

スタイルシートの策定はW3C (World Wide Web Consortium) というWeb技術の標準化団体が中心になって行っており、現在はLevel2という2代目の勧告が最新のものとなっています。

勧告では、各プロパティの働きなどのほかに、以下のような内容を定めています。

適用

プロパティを適用できる要素の種類

ブロックレベル要素、インラインレベル要素などについてはp.4を参照してください。また、displayプロパティ（p.202参照）によって表示形式を変更して、インラインレベル要素にのみ適用できるプロパティをブロックレベル要素に適用するといった方法もあります。

デフォルト

プロパティのデフォルト値（初期値）

継承

設定したプロパティの内容が子要素など下位の要素に継承されるかどうか

継承についてはp.44を参照してください。

下記の表は、各プロパティのW3Cにおける定義内容です。ブラウザが独自に拡張しているプロパティについても同様に、ブラウザが定義している内容を掲載しています。

各ブラウザはこれらをもとに対応をはかるわけですが、必ずしも定義通りに実装しているわけでなく、ブラウザの種類やバージョンによっては定義と異なる場合があります。

プロパティ	適用	デフォルト	継承	独自拡張
background	すべての要素	定義されていない	しない	-
background-attachment	すべての要素	scroll	しない	-
background-color	すべての要素	transparent	しない	-
background-image	すべての要素	none	しない	-
background-position	ブロックレベル要素、置換要素	0% 0%	しない	-
background-position-x	すべての要素	0%	しない	IE独自
background-position-y	すべての要素	0%	しない	IE独自
background-repeat	すべての要素	repeat	しない	-
behavior	すべての要素	なし	しない	IE独自
border	すべての要素	各プロパティのデフォルト	しない	-
border-bottom	すべての要素	各プロパティのデフォルト	しない	-

プロパティ	適用	デフォルト	継承	独自拡張
border-bottom-color	すべての要素	**color** プロパティの値	しない	-
border-bottom-style	すべての要素	各プロパティのデフォルト	しない	-
border-bottom-width	すべての要素	**medium**	しない	-
border-collapse	テーブル要素	**collapse**	する	-
border-color	すべての要素	各プロパティのデフォルト	しない	-
border-left	すべての要素	各プロパティのデフォルト	しない	-
border-left-color	すべての要素	**color** プロパティの値	しない	-
border-left-style	すべての要素	各プロパティのデフォルト	しない	-
border-left-width	すべての要素	**medium**	しない	-
border-right	すべての要素	各プロパティのデフォルト	しない	-
border-right-color	すべての要素	**color** プロパティの値	しない	-
border-right-style	すべての要素	各プロパティのデフォルト	しない	-
border-right-width	すべての要素	**medium**	しない	-
border-spacing	テーブル要素	**0**	する	-
border-style	すべての要素	**none**	しない	-
border-top	すべての要素	各プロパティのデフォルト	しない	-
border-top-color	すべての要素	**color** プロパティの値	しない	-
border-top-style	すべての要素	各プロパティのデフォルト	しない	-
border-top-width	すべての要素	**medium**	しない	-
border-width	すべての要素	各プロパティのデフォルト	しない	-
bottom	注1	**auto**	しない	-
caption-side	キャプション要素	**top**	する	-
clear	ブロックレベル要素	**none**	しない	-
clip	ブロックレベル要素、置換要素	**auto**	しない	-
color	すべての要素	ブラウザ依存	する	-
content	**:before** 擬似要素、**:after** 疑似要素	空文字	しない	-
cursor	すべての要素	**auto**	する	-
direction	すべての要素	**ltr**	する	-
display	すべての要素	**inline**	しない	-
empty-cells	テーブル要素、セル要素	**show**	する	-
filter	すべての要素	なし	しない	IE 独自
float	すべての要素（注1、注2を除く）	**none**	しない	-
font	すべての要素	各プロパティのデフォルト	する	-
font-family	すべての要素	ブラウザ依存	する	-
font-size	すべての要素	**medium**	する	-
font-style	すべての要素	**normal**	する	-
font-variant	すべての要素	**normal**	する	-

プロパティ	適用	デフォルト	継承	独自拡張
font-weight	すべての要素	**normal**	する	-
height	すべての要素（注3）	**auto**	しない	-
ime-mode	**input** 要素、**textarea** 要素	**auto**	する	IE独自
layout-grid	ブロックレベル要素	各プロパティのデフォルト	する	IE独自
layout-grid-char	ブロックレベル要素	**none**	する	IE独自
layout-grid-line	ブロックレベル要素	**none**	する	IE独自
layout-grid-mode	ブロックレベル要素	**both**	する	IE独自
layout-grid-type	ブロックレベル要素	**loose**	する	IE独自
left	注1	**auto**	しない	-
letter-spacing	すべての要素	**normal**	する	-
line-break	ブロックレベル要素、テーブル要素、セル要素	**normal**	する	IE独自
line-height	すべての要素	**normal**	する	-
list-style	リストアイテム要素	定義されていない	する	-
list-style-image	リストアイテム要素	**none**	する	-
list-style-position	リストアイテム要素	**outside**	する	-
list-style-type	リストアイテム要素	**disc**	する	-
margin	すべての要素	定義されていない	しない	-
margin-bottom	すべての要素	**0**	しない	-
margin-left	すべての要素	**0**	しない	-
margin-right	すべての要素	**0**	しない	-
margin-top	すべての要素	**0**	しない	-
overflow	ブロックレベル要素、置換要素	**visible**	しない	-

注1　position プロパティで strict 以外の値が指定された要素
注2　スタイルシートで自動生成される要素
注3　非置換インライン要素とテーブル列（col 要素、colgroup 要素）を除く
注4　一部を除く
注5　スクロールバーを生成する要素
注6　ブラウザと文字の表記方向による
注7　auto（IE6 以降）、below（IE5.5）
注8　非置換インライン要素と tr 要素、 thead 要素、tbody 要素、tfoot 要素を除く
注9　フォームの部品など一部の要素を除く

6 APPENDIX

スタイルシート乗換一覧

同じ効果が得られるHTMLタグとスタイルシート（CSS）の併記一覧です。

HTMLタグの表記をCSSに改めたい場合、またはCSSの記述をHTMLタグに改めたい場合に参照してください。ただし、デザインに関する指定はCSSでの指定が推奨されています。

改行しないで表示させたい

HTML表記　**<nobr>～</nobr>**

CSS表記　**☆ { white-space: nowrap }**

☆——セレクタ

入力したとおりに表示したい

HTML表記　**<pre>～</pre>**

CSS表記　**☆ { white-space: pre }**

☆——セレクタ

上付き文字・下付き文字を指定したい

HTML表記　**[～]**（上付きの場合）

_～（下付きの場合）

CSS表記　**☆ { vertical-align: ★ }**

☆——セレクタ

★——super（上付きの場合）、sub（下付きの場合）

テキストを点滅表示させたい

HTML表記　**<blink>～</blink>**

CSS表記　**☆ { text-decoration: blink }**

☆——セレクタ

背景色を指定したい

HTML表記　**<body bgcolor="★">～</body>**

CSS表記　**body { background-color: ★ }**

★——色

背景に画像を使いたい

HTML表記　<body background="★">～</body>

CSS表記　body { background-image: ★ }

★――画像のURL

背景画像を固定したい

HTML表記　<body background="☆" bgproperties="fixed">～</body>

CSS表記　body { background-image: ☆; background-attachment: fixed }

☆――画像のURL

テキストの色を指定したい（文書全体）

HTML表記　<body text="★">～</body>

CSS表記　body { color: ★ }

★――色

テキストの色を指定したい（未参照リンクの文字色）

HTML表記　<body link="★">～</body>

CSS表記　a:link { color: ★ }

★――色

テキストの色を指定したい（リンクをクリックしたときの文字色）

HTML表記　<body alink="★">～</body>

CSS表記　a:active { color: ★ }

★――色

テキストの色を指定したい（参照済みリンクの文字色）

HTML表記　<body vlink="★">～</body>

CSS表記　a:visited { color: ★ }

★――色

テキストの色を部分的に変更したい

HTML表記　～

CSS表記　☆ { color: ★}

☆――セレクタ

★――色

見出しの位置を指定したい

HTML表記　<h☆ align="★">～</h>

CSS表記　h☆ { text-align: ★ }

☆――1～6

★――left、center、right

段落の位置を指定したい

HTML表記　**<p align="★">～</h>**

CSS表記　**p { text-align: ★ }**

★──left、center、right

まとめて位置を指定したい

HTML表記　**<div align="★">～</div>**

CSS表記　**div { text-align: ★ }**

★──left、center、right

センタリングしたい

HTML表記　**<center>～</center>**

CSS表記　**☆ { text-align: center }**

☆──セレクタ

ページのマージンを指定したい

HTML表記　**<body leftmargin="★">～</body>**

CSS表記　**body { margin-left: ★ px }**

★──マージン幅

※ 同様に **topmargin、rightmargin、bottommargin** についても **margin-top、margin-right、margin-bottom** で表現できます

フォントサイズを指定したい

HTML表記　**～**

CSS表記　**☆ { font-size: ▲ }**

★──1～7

☆──セレクタ

▲──数値やキーワード

フォントの種類を指定したい

HTML表記　**～**

CSS表記　**☆ { font-family: ★,★… }**

☆──セレクタ

★──フォント名

フォントスタイルを指定したい（太字）

HTML表記　**～**

CSS表記　**☆ { font-weight: bold }**

☆──セレクタ

フォントスタイルを指定したい（斜体）

HTML表記　<i> ～ </i>

CSS表記　☆ { font-style: italic }

☆――セレクタ

フォントスタイルを指定したい（取り消し線）

HTML表記　<strike> ～ </strike> または <s> ～ </s>

CSS表記　☆ { text-decoration: line-though }

☆――セレクタ

フォントスタイルを指定したい（等幅）

HTML表記　<tt> ～ </tt>

CSS表記　☆ { font-family: monospace }

☆――セレクタ

フォントスタイルを指定したい（下線）

HTML表記　<u> ～ </u>

CSS表記　☆ { text-decoration: underline }

☆――セレクタ

フォントスタイルを指定したい（大きめの文字、小さめの文字）

HTML表記　<big> ～ </big>（大きめの文字）

<small> ～ </small>（小さめの文字）

CSS表記　☆ { font-size: ★ }

☆――セレクタ

★――larger（大きめの文字）、smaller（小さめの文字）

リストのマークを変更したい

HTML表記　<ul type="★"> ～

CSS表記　ul { list-style-type: ★ }

★――リストのマーク

画像のサイズを指定したい

HTML表記　<img src="☆" width="★" height="▲">

CSS表記　img { width: ★; height: ▲ }

☆――画像のURL

★――幅

▲――高さ

画像に枠線をつけたい

HTML表記　**<img src="☆" border="★">**

CSS表記　**img { border: △ ◇ ★ px }**

☆――画像のURL

△、◇――枠線のスタイルと色

★――枠線の太さ

テキストとの並び方を指定したい

HTML表記　**<img src="☆" align="★">**

CSS表記　**img { vertical-align: ★ }**

☆――画像のURL

★――**top、middle、bottom**

画像にテキストを回り込ませたい

HTML表記　**<img src="☆" align="★">**

CSS表記　**img { float: ★ }**

☆――画像のURL

★――**left、right**

回り込みを解除したい

HTML表記　**<br clear="★">**

CSS表記　**br { clear: ★ }**

★――**left、right**

回り込みの際の画像とテキストの間隔を指定したい

HTML表記　**<img src="☆" vspace="★" hspace="◆" align="◇">**

CSS表記　**img { margin: ★ px ◆ px; float: ◇ }**

☆――画像のURL

★――上下方向の間隔

◆――左右方向の間隔

★――**left、right**

テーブル枠線の幅を設定したい

HTML表記　**<table border="★"> ～ </table>**

CSS表記　**table { border:△ ◇ ★ px }**

★――枠線の太さ

△、◇――枠線のスタイルと色

テーブルのサイズを設定したい

HTML 表記　**<table width="★" height="▲"> ～ </table>**

CSS 表記　**table { width: ★; height: ▲ }**

★──幅

▲──高さ

セルのサイズを指定したい

HTML 表記　**<th width="★" height="▲"> ～ </th>**

CSS 表記　**th { width: ★; height: ▲ }**

★──幅

▲──高さ

※ **<td>** タグについても同様です

セル内のテキストの位置を指定したい

HTML 表記　**<tr align="★" valign="☆"> ～ </tr>**

CSS 表記　**tr { text-align: ★; vertical-align: ☆ }**

★──**left、center、right**

☆──**top、middle、bottom**

※ **<th>、<td>** タグについても同様です

セルの間隔やマージンを変更したい

HTML 表記　**<table cellspacing="★" cellpadding="☆"> ～ </table>**

CSS 表記　**table { border-spacing: ★ px }**

td {padding: ☆ px;}

★──枠線の太さ

☆──マージン

7 APPENDIX

プロパティインデックス

スタイルシートのプロパティで検索するインデックスです。

8 APPENDIX

値インデックス

スタイルシートの値からプロパティのページを逆引きするインデックスです。

h

i

j

k

l

m

n

o

p

r

s

用語インデックス

スタイルシートの効果をあらわす用語や、関連する言葉を検索するインデックスです。

記号

数字

英字

き

く

け

こ

さ

し

わ

Information

Web辞典シリーズのホームページでは、本書サンプルソースのダウンロードのほか、カラーチャート正誤表などをホームページにて掲載予定です。ぜひご利用ください。

http://www.shoeisha.com/book/pc/dic/

Staff

装丁
萩原 美和（VAC Creative）

ビジュアルインデックス サンプル作成
フクイ メグミ

スタイルシート辞典 第3版

2002年 7月12日　初版第1刷発行

著　者　（株）アンク
発行人　速水 浩二
発行所　株式会社 翔泳社
http://www.shoeisha.com
印刷・製本　凸版印刷株式会社

ISBN4-7981-0271-7　Printed in Japan

第1部　スタイルシートの基礎

第2部　スタイルシートリファレンス

テキスト/TEXT
フォント/FONT
背景/BACKGROUND
ボックス/BOX
配置/POSITIONING
リスト/LIST
テーブル/TABLE
フィルタ/FILTER
その他/OTHER

付録

Webページカラーチャート
色の基礎知識
Web配色サンプル
ビジュアルインデックス
適用・デフォルト・継承一覧
スタイルシート乗換一覧
プロパティインデックス
値インデックス
用語インデックス

9784798102719

1923055018006

ISBN4-7981-0271-7

C3055 ￥1800E

株式会社 翔泳社
定価：本体 1,800円＋税

スタイルシート辞典

第3版

(株)アンク著